U.S.A. $3.50
CANADA $4.50

NOW 520ST Special Section

Antic®

The ATARI® Resource

SEPTEMBER 1985 VOLUME 4, NUMBER 5

POWER PROGRAMMING

Type in Revision C BASIC
First 130XE Program
520ST Memory Map

Hackers: What are they?

PRO TECHNIQUES

- Page flipping
- Mirrored displays
- Fine scrolling
- 16-bit sound

INSIDE: 10 Type-In Programs



THE CREATION OF A NEW BREED



The GCP Network: More Communications, More Graphics, LESS Hassle!

The GCP City:
The entry to worlds of adventure and challenge.



The Game Floor:
Choose your favorite and match wits with others.



Lords of Space:
Explore and conquer your way to galactic supremacy.



FEATURES: Joystick-driven; no commands to learn! Private mail, bulletin boards, program library, file transfers, automatic logons, User Group support, local phone call access from most cities, no network surcharge. SYSTEM REQUIREMENTS: 48K Atari, 1 disk drive, modem.

For more information contact:

GCP GAMES COMPUTERS PLAY, INC.

112 East Market Street, York, PA 17401 / 717-848-2660

For COMMODORE 64 ATARI APPLE 11e/c AND IBM PC

Imagine...

A WORD PROCESSING PROGRAM!
AN INFORMATION MANAGEMENT PROGRAM!
A TELECOMMUNICATIONS PROGRAM!

ALL THREE PROGRAMS, ON ONE DISKETTE!, FOR ONLY $49.95!*

"Quite simply the best! The highest rating possible...the package should be part of every (computer) library." — ANALOG COMPUTING

"Russ Wetmore has done an EXCELLENT job! The program is flexible, powerful and very easy to use. $49.95 buys a heck of a lot of program." — REVIEW BY ARTHUR LEYENBERGER

"Performance: ★ ★ ★ ★ (Excellent) Value: ★ ★ ★ ★ (Excellent) This three-in-one package is a bargain . . . one of the finest values on the market."
FAMILY COMPUTING

HOMETEXT WORD PROCESSOR.
HOMEFIND INFORMATION MANAGER.
HOMETERM TELECOMMUNICATIONS

Together they are **HomePak**: the three most important and most useful home computer applications in one integrated system — on one diskette!

The reviewers are unanimous: any one of these programs **alone** is well worth the price. So you're getting **three times the computing power**, with this exceptionally easy to use package:

- **all commands in simple English; no complex computer jargon, no obscure instructions**
- **all key commands are immediately available on the screen menu; additional commands can be called up for the more experienced user**
- **to help you, system status is displayed right on the screen**

And it's easy to use the three programs together. For example, in the "Merge" mode, you can take data stored in HOMEFIND and print letters and labels using HOMETEXT. Or, use **HOMETEXT** to write reports based on information you've called up via **HOMETERM**.



BATTERIES INCLUDED

"The Energized Software Company!"

30 Mural Street
Richmond Hill, Ontario
L4B 1B5 CANADA
(416) 881-9941
Telex: 06-21-8290

17875 Sky Park North, Suite P
Irving, California
USA 92714
(416) 881-9816
Telex: 509-139

WRITE TO US FOR FULL COLOUR CATALOGUE of our products for COMMODORE, ATARI, APPLE and IBM SYSTEMS

FOR TECHNICAL SUPPORT OR PRODUCT INFORMATION PLEASE PHONE (416) 881-9816

*MANUFACTURERS SUGGEST U.S. LIST PRICE

Join the Atari® Safari
of Software and Books



Get Out of the Thick of Things

If too many bills are piling up around you, *The Home Accountant*™ will put you back on safe ground. *The Home Accountant* handles cash, checkbooks and credit cards. Custom retrieval allows you to print all the information you need, including net worth, balance statements and checks.

Learn to Create Wild Graphics

Create your own electronic art with *Atari Color Graphics: A Beginner's Workbook*. So simple to follow, even a complete novice can program stunning graphic displays by following a few easy steps.

Program Ferocious Games

Atari Graphics and Arcade Game Design will teach you how to design quality games on the Atari. A series of flow charts, text and commented code explain everything in detail. Five complete games show you how it's done.

Conquer Your Computer

If you're wondering what to do with your Atari now that you've played countless games, get *The Atari User's Encyclopedia*. This book tells everything you need to know for creative and satisfying use of your computer.

Put an end to the computer software and book hunt today. **Write or call our Product Information department for the dealer in your area.**

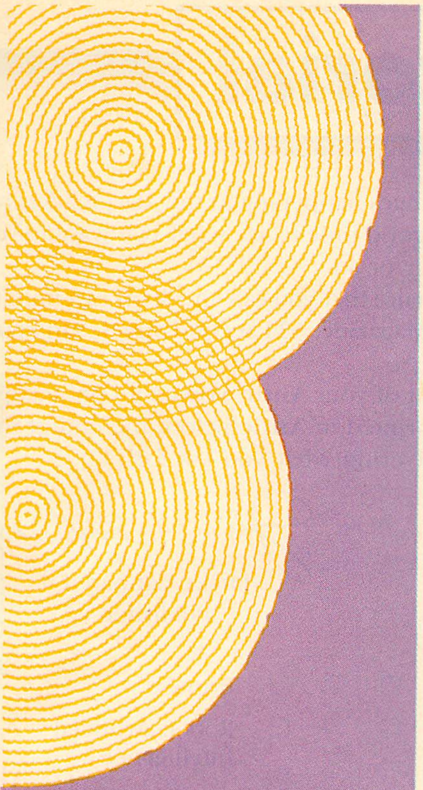
Arrays, Inc./Continental Software
Arrays, Inc./The Book Division
Attn: Product Information
11223 S. Hindry Ave., Los Angeles, CA 90045
(213) 410-3977

The Home Accountant hardware requirements: 48K Atari 800/800XL, one or two disk drives, printer (132 Col. optional)

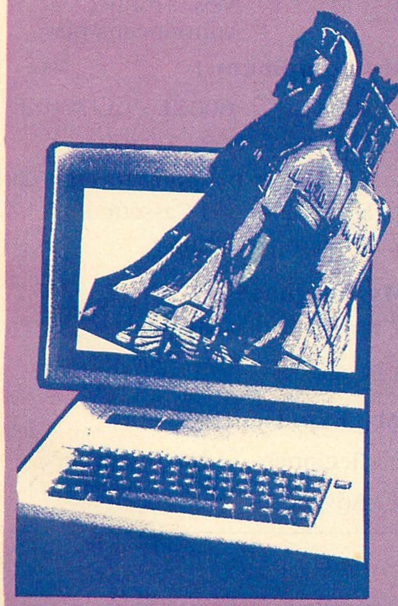
NEW! ST SECTION page 15

Antic®

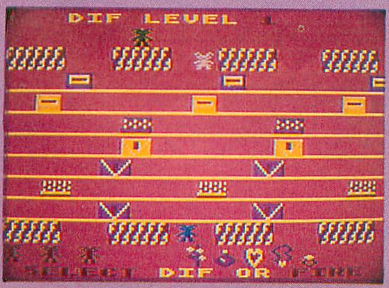
The ATARI® Resource SEPTEMBER 1985, VOLUME 4, NUMBER 5



Plotters for your Atari 42



Hackers forever! 45



Crickets in love 13

FEATURES

SOFTWARE LIBRARY

ST SECTION

DEPARTMENTS

General Offices
& Catalog Customer Service
(415) 957-0886
Subscription Customer Service
(614) 383-3141
Antic, P.O. Box 1919, Marion, OH 43306
Advertising Sales (415) 661-3400
Garland & Associates (617) 749-5852
Credit Card Subscriptions & Catalog Orders
outside California (800) 227-1617 ext. 133
inside California (800) 772-3545 ext. 133

September 1985, Volume 4, Number 5
Antic—The Atari Resource is published twelve times per year by Antic Publishing. Editorial offices are located at 524 Second Street, San Francisco, CA 94107. ISSN 0745-2527. Second Class Postage paid at San Francisco, California and additional mailing offices. POSTMASTER: Send address change to **Antic**, P.O. Box 1919, Marion, OH 43306.

Editorial submissions should include program listing on disk or cassette, and text file on media and paper if text was prepared with a word processor. Media will be returned if self-addressed stamped mailer is supplied. **Antic** assumes no responsibility for unsolicited editorial material.

editorial

With all the dramatic changes happening in the Atari market recently, this seems like a good time to ask what *you* want to see in **Antic** for the next year or so.

Here is a short questionnaire that you can check off in a minute or two. Then mail it to: Survey, **Antic** Magazine, 524 Second Street, San Francisco, CA 94107.

If you don't want to tear this page out of your **Antic**, you could photocopy it. The results of your voting will be printed in **Antic** before the end of the year. So there'll be all of 1986 for delivering what you ask for!

Thanks for your help.

James Capparell
Publisher

ANTIC READER SURVEY

Please check off whether you want these topics covered in **Antic** MORE than they are now, LESS than now, or the SAME amount as now.

MORE	LESS	SAME	
☐	☐	☐	**Type-In Programs:**
☐	☐	☐	Practical Applications
☐	☐	☐	Financial
☐	☐	☐	Games
☐	☐	☐	Educational
☐	☐	☐	Graphics
☐	☐	☐	Music/Sound
☐	☐	☐	Utilities/Tutorials
☐	☐	☐	Beginners' Easy
☐	☐	☐	**Articles Without Programs:**
☐	☐	☐	Atari News
☐	☐	☐	Interviews
☐	☐	☐	ST Section
☐	☐	☐	Telecommunications
☐	☐	☐	Printers
☐	☐	☐	Robots/Artificial Intelligence
☐	☐	☐	Cryptography
☐	☐	☐	Product Reviews
☐	☐	☐	New Product Announcements

My Equipment:

☐ 800XL ☐ 600XL ☐ 1200XL
☐ 130XE ☐ 800 ☐ 400
☐ 520ST (Or planning to buy 520ST)
☐ Disk Drive ☐ Cassette
☐ Modem ☐ Printer

My Languages:

☐ BASIC ☐ Assembler ☐ ACTION!
☐ Logo ☐ Forth ☐ C

My Experience Level:

☐ Beginner ☐ Intermediate
☐ Advanced

Additional topics I would like:

What I like BEST about Antic:

What I like LEAST about Antic:

i/o board

TYPO II TRIBULATIONS

I wrote to **Antic** about TYPO II, which I couldn't get to run, despite "50 tries." I had practically lost faith in you! All was rectified in your June, 1985 HELP letter "Checked 100 Times." Using the suggestions in your answer, I then typed in TYPO II again and found that it worked perfectly on the first try.

Charles Hostetler
Detroit, MI

Glad we could help. As we stated in June, it can be confusing when you're getting started typing in magazine programs. We recommend that newcomers start with shorter programs and be sure to read all the typing hints at the start of the Software Library listing section.—ANTIC ED

MYSTIFYING MODULE

Here's a mystery for you! I am the happy owner of a 1090 XL Expansion Module. That's right, a 1090. Does anyone have any information on this device? It came right off the assembly line without documentation.

Stephen Warn
East Helena, MT

The Atari 1090 XL was an intriguing idea that never got to the market. It was designed by a Fellow of the Atari Institute, Bill Steuben, to give XL's compatibility with other microcomputers such as the IBM PC. The project was stopped a week after being exhibited at the June, 1984 CES. Software and plug-in boards that were to accompany it were never completely developed.

Antic *now owns a 1090 and it makes a good paperweight.*—ANTIC ED

COVER COMMENTS

I am disappointed that C. A. Castravelli's letter called your covers childish. I think they're great and my schoolmates like them also.

Michael De Fong
Tyndall, Manitoba

PRINTSHOP PRINTERS

Will Broderbund's Print Shop software work with my Okimate 10 printer? I hope you can help.

Todd Hartmann
Edina, MN

Broderbund says that Print Shop will not work with the Okimate 10. A ***complete*** *list of compatible printers appears on the back of the box in the lower left hand corner. Broderbund requested us to emphasize that the Commodore 6502 and 802 printers also do not work with Print Shop.*—ANTIC ED

SYNFILE+ PRINT STYLE

I have discovered that it is possible to change SynFile+ print styles from inside the program.

Create a separate print file using the lookup field. Enter printer escape codes as records in this field of your file. When the record is printed the printer code is sent to the printer. This file can then be closed and another file loaded with your data now printed in your selected font. It will also work with the conditional field, for changes in print styles in individual records.

Bob Stirling
Yukon, Canada

MODEM CHOICE

I need some help buying a modem. I have narrowed my choice down to two of them, but am having difficulty picking one. Please help.

Chuck Ryckman
Brown City, MI

There are many modems on the market and the choice can be confusing (especially if you own an 850 interface). Next issue of ***Antic*** *reviews some good new modems at various price levels. See if there's anything you like.*—ANTIC ED

ONE-ON-ONE 130XE

New Atari 130XE computers may have difficulty loading Electronic Arts One-on-One game. I have found that the following procedure works:

1. Load in Fix XL or XL Translator (Side B) holding down [OPTION].
2. Following instructions, replace the translator disk with One-on-One disk and press [SELECT].
3. When the drive pauses, quickly toggle computer off and on, while holding [OPTION] down. Continue to hold [OPTION] down during the rest of the loading process, until the demo screen comes up.

Kim Ellison
3E Software & Systems
Hayward, CA

Thanks for the advice, Kim. This is the only program ***Antic*** *knows about so far that has any trouble running on the 130XE.*—ANTIC ED

SPEAKING IN TONGUES

What different programming languages are available for the Atari and who markets them?

Jeffrey Velasquez

Not all the languages produced for the Atari are currently available. Because you may be able to find some of these "out-of-print" languages, we've included them in the following list. There are several BASICs from Atari itself. BASIC XL, ACTION! and C/65 are produced by Optimized Systems Software. Another version of C is called Deep Blue C, and is available from the Antic Arcade Catalog, as is FORTH. There are two versions of Pascal—Draper and I.S.O., neither of which is presently available. (I.S.O. Pascal and an Atari LISP were distributed by the old Atari Programmers Exchange.) Atari distributes Logo and PILOT. Finally, there are several assemblers around, including MAC/65 from OSS, Atari's Assembler Editor and Macro Assembler.—ANTIC ED

antic online

THUMBS UP

We'd like to thank the dozens of people who responded to our plea for help with **Hitchhiker's Guide to the Galaxy** (reviewed May '85). Not only were the responses plentiful, most were written in the true style of Infocom—hints and more hints, with carefully concealed answers should we fail with the clues. —Jack Powell and Michael Ciraolo of **Antic**.

NOT TAKING IT

(Russell Casey of Lexington, KY sent along a copy of this letter from the president of Electronic Arts. It's yet another response to the **Antic** *May, 1985 editorial urging readers to demand that software companies release Atari versions of their hit titles.)*

Personally I am a big Atari fan and my 800 is my favorite CPU among the five I have at home. We do have about four more releases planned for the Atari this year.

My figures indicate that perhaps 750,000 Atari CPU's have been sold to date, but only about 200,000 of those have disk drives. Many CPU's were sold by stores like Sears and K-Mart when no drives were even available. The "new" Atari claims to be selling lots of drives, and that's great.

Unfortunately for Electronic Arts and Atari users, most retailers are uncertain about the future of Atari and therefore are still trying to sell the old software they have, rather than bring in new titles. There is little incentive for us to develop them. Let them know that you are ready to buy new titles for your Atari!

Trip Hawkins
President,
Electronic Arts

We disagree with Mr. Hawkins only about his opinion that two-thirds of Atari owners don't have disk drives. ***Antic*** *believes that with drive prices dropping below $200 last year, virtually all active Atari users are now upgraded to disk. Certainly we rarely see a program submitted here on cassette these days.*
—ANTIC ED

NEW IN AUGUST

Type GO ANTIC when you log onto CompuServe in August. You'll find the first lessons from Chris Crawford's Assembly Language tutorial now on ANTIC ONLINE, in the Worldwide Users Network (WUN) pages.

Each month a new AL lesson by famed games designer Crawford will be uploaded. Here is the complete course:

1. 6502 Assembler
2. Arithmetic
3. Logic
4. Branching
5. Index Registers
6. Subroutines
7. Interrupts
8. Advanced Topics

Representatives of users groups affiliated with WUN have been uploading their suggestions for future WUN programs. You can follow the ongoing discussion in the WUN section of ANTIC ONLINE, as well as keeping up with the latest WUN news.

Atari's Sig Hartmann, president of the AtariSoft Division, has agreed to serve on the WUN board of directors. Hartmann has a long record of service to users groups, Commodore users newsletters carried grieving headlines when he left to join Jack Tramiel at Atari.

CUSTOMER SERVICE

A complete Antic Arcade Catalog customer service system is now in place on ANTIC ONLINE. You can follow the Antic Central menu prompts and upload your queries for quick response via email.

If you're not a CompuServe subscriber yet, see your local computer dealer or phone (800) 848-8199 for sign-up information. Ohioans phone (614) 457-0802. You can access ANTIC ONLINE evenings at 300 baud for only the standard CompuServe hourly $6 rate.

Worldwide™ Users Network

help!

GUESS THAT LINE

If Yellow Submarine seems a little out of tune in "Guess That Song" (July, 1985), it's probably because some of the lines were a little blurry and hard to type in. Below is a clear printout of the problem lines.

```
4000 DATA 15,8,11y1▚▌▀Y
6Y,AAABBBBABA,"Sunshine
 of Your Love"
4010 DATA 10,4,▚◤y▚▀▚6▚
▚▀,BAHABAIBAB,"Yellow S
ubmarine"
4020 DATA 16,4,▀66▚▚r1◆
[◆,AAHAAAAAAB,"Yesterda
y"
4130 DATA 20,7,YY▬▚▚HHU
UY,AAAABABABA,"The Blue
 Danube"
4140 DATA 6,5,▀▀▀L666Y▀
▀,CCCNCCCLBB,"Beethoven
's Fifth Symphony"
4270 DATA 6,20,▀▀◤y▀▀6▬
▀▀,CCBBCCBBCC,"Mission
Impossible"
4280 DATA 10,5,A▚1r▚▬▚◆
◆L,CAFBAAAFAJ,"Star Tre
k"
4310 DATA 9,5,▀ASA▀y◆1y
A,BACCCFBACC,"The Star-
Spangled Banner"
4350 DATA 30,3,A▬▀yYA▬▚
▚S,AAAEAAABCA,"Autumn L
eaves"
```

SOUND EFFECTS LIBRARY

(And introduction to SOUND command)

by TIMOTHY BANSE

Start creating spectacular music and sound effects for your own BASIC programs. This tutorial introduces beginners to the Atari's SOUND command. There's also a BASIC program that demonstrates 17 Atari sound effects. It works on all 8-bit Atari computers of any memory size, with disk or cassette.

The SOUND (SO.) statement has six parameters (components).

```
100 SOUND 0,255,10,8
```

In the above example, those components are the line number 100, the SOUND command which can be abbreviated as SO., and four values representing: one of the four **voices, pitch, distortion** and **volume**.

There are four voice channels, 0,1,2, and 3, each needing its own set of pitch, distortion and volume values. Furthermore, it's possible to play just one voice at a time, or any combination.

THREE VALUES

Pitch determines the high or low frequency of the sound. Pitch can be represented by any number between 0 and 255. The bigger the number, the lower the note, with 255 being the lowest possible note. The frequency range spans just over three octaves.

Distortion can be regulated by plugging in any value between 0 and 15. The values 10 and 14 generate pure notes, a clean sound. Manipulating the purity or distortion of the tone is a powerful tool for controlling sound effects. Here's where experimentation will really pay off. Don't be hesitant to replace a pure tone's "10" with a buzzsaw "6."

Volume indicates the loudness of the sound you hear from your video speaker. Again, use any number between 0 and 15. Zero is silent while 15 is the loudest. 8 is a good choice when your living accommodations require you to consider the eardrums of your neighbors. Note that the sum of all the volumes should total less than 32.

One of the best ways to improve on the tonal quality of a particular sound effect is to decrement (reduce) the volume. Start out with the volume at 8 and reduce it to 1 or 0. You'll see a few examples in the Sound Effects Library.

MAKING MUSIC

But first, let's learn about one of the important programming tools available to the SOUND command. It's the FOR-NEXT loop. Coupled with numbered DATA that represents musical tones, you can program a musical rendition of whatever melody you've composed. Here's just such a program:

```
100 VOICE=0
110 PITCH=0
120 DISTORTION=10
130 VOLUME=8
140 FOR COUNT=1 TO 5
150 READ PITCH
160 SOUND VOICE,PITCH,DIS
TORTION,VOLUME
200 FOR DELAY=1 TO 100:NE
XT DELAY
210 NEXT COUNT
220 SOUND 0,0,0,0
230 DATA 200,190,20,34,40
```

The program reads data statements and converts them into musical notes. Line 140 controls the loop that will read each one of the data numbers. Line 150 reads that number and line 160 plays the note. Line 200 is a delay loop, it plays the musical note for a count of 100 before letting the program go on to the next line. If you want a slower tune, increase the count to 1,000. Or speed up the tempo by shortening the delay count to 50.

It's easy to combine all four voices to create chords. With just a couple of additions we can put chords into the previous music program:

```
100 FOR COUNT=1 TO 5
110 READ ONE,TWO,THREE,FO
UR
120 SOUND 0,ONE,10,8
130 SOUND 1,TWO,10,8
140 SOUND 2,THREE,10,8
150 SOUND 3,FOUR,10,8
160 FOR DELAY=1 TO 100:NE
XT DELAY
170 NEXT COUNT
180 DATA 40,34,30,200
190 DATA 200,56,99,234
200 DATA 156,162,170,190
210 DATA 45,45,56,175
220 DATA 200,190,34,40
```

Even though we've just added a few lines, the results will be subtantially different. First of all, we changed how we read the musical notes. Now we're reading four musical notes, stored as data statements, all in the same program line.

continued on next page

Originally, the tune played the notes stored as 40, 34, 30, 200 . Now each voice will in turn take one of the data numbers as its first musical note. Either of these two programs is easy to modify to your own tastes and requirements.

The table below shows approximate musical note values. The higher the number, the lower the pitch:

PITCH	NUMERIC VALUE	
C	60	
B	64	
A#/Bb	68	
A	72	
G#/Ab	76	
G	81	
F#/Gb	85	High Notes
F	91	
E	96	
D#/Eb	102	
D	108	
C#/Db	114	
C	121	Middle C
B	128	
A#/Bb	136	
A	144	
G#/Ab	153	
G	162	
F#/Gb	173	
F	182	
E	192	
D#/Eb	204	Low notes
D	217	
C#/Db	230	
C	243	

SOUND EFFECTS

Atari audio isn't limited to making beautiful music. Sound effects can breathe life into a game or liven up an applications program.

Suppose you've just written a useful but dull checkbook balancing program and you want to make it more interesting. You could use the SETCOLOR command to change the screen border to red if the balance drops below one cent. Then underscore the visual with a sound effect like a bronx cheer or a siren.

Here is a library of sound effects to help get you started. Type in Listing 1, SOUND.BAS, check each line with TYPO II and SAVE a copy. When you RUN the program, you'll see a menu of 17 choices which you can select by number.

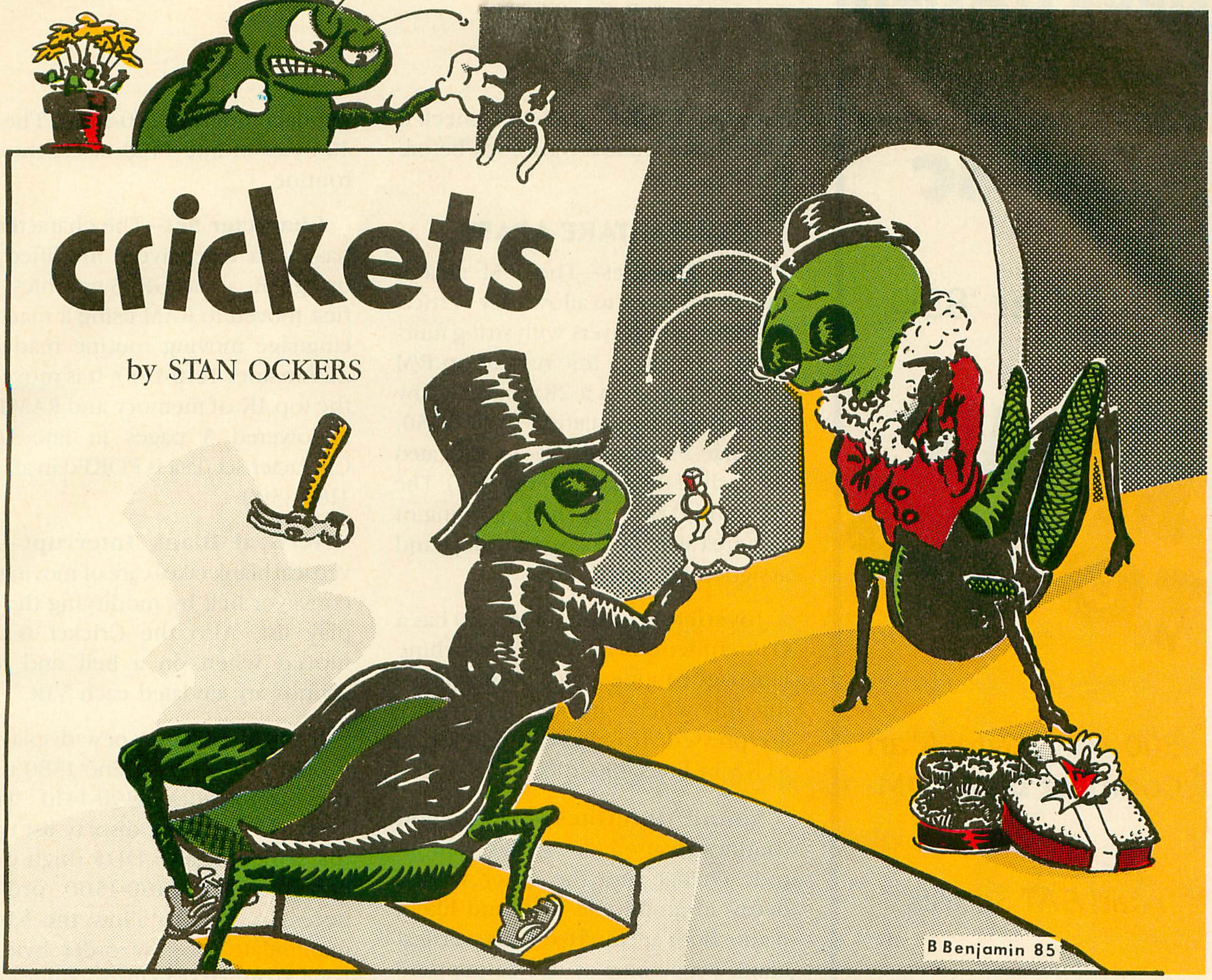
The program lines for each sound start off with a REM description. Use the sounds as is, or feel free to experiment with any of them. Plug the sounds into your own programs as needed.

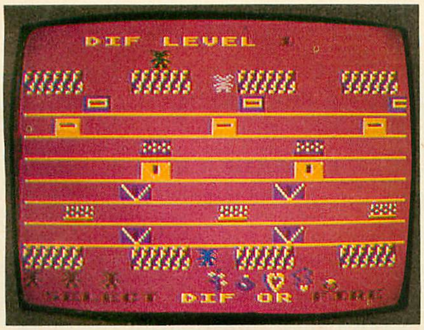
*Timothy Banse is a first-time **Antic** author from Coralville, Iowa.*

Listing on page 68

crickets

by STAN OCKERS





Meet Clyde the lovesick cricket in this fun-filled obstacle jumping game. Clyde must make his way through a booby-trapped factory to get to his true love, Cynthia. Crickets is from the master of Atari public domain programming, Stan Ockers. This BASIC program works on all Atari computers having 32K, with disk or cassette.

Clyde Cricket lives in a factory and is deeply enamored of Cynthia Cricket. He risks his life to bring her gifts, hoping to eventually win her feeler in marriage.

In order to deliver his gifts to the fair Cynthia, poor Clyde must jump between moving conveyor belts and avoid getting crushed by the objects upon the belts. He must also avoid touching any of the factory walls, floors, or ceilings—they've all been sprayed with a fierce anti-cricket poison.

To woo Cynthia successfully, Clyde must bring her flowers, perfume, candy, a necklace and finally. . .a ring. Each gift must be picked up at the center of the factory's lowest level and presented to Cynthia at her home on the uppermost level.

If Clyde fails, his three amorous brothers are ready to take his place. These Cricket brothers don't have to start at the beginning—they simply take over the current gift delivery. One last problem for Clyde—a jealous rejected suitor makes things difficult by throwing various objects from the top of the screen.

To make Clyde jump, you must move the joystick in the correct direction and simultaneously press the fire button.

You may use the [SELECT] button to choose a level of difficulty, the top ones border on the impossible. If you don't want so many, you can change the 7 in line 480 to a lower number.

Anyhow, time to get started. Type

continued on next page

in Listing 1, CRICKETS.BAS, check it with TYPO II and SAVE a copy before you RUN it.

PROGRAM TAKE-APART

Player Missiles—The P/M area is made of strings, to allow easy vertical movement of players with string functions. The single line resolution P/M area is located on a 2K boundary by the string manipulation in line 150. Separate 256-byte strings are allocated for each of the players (PO$-P3$). The 1K unused space at the beginning of the area is used for screen data and assigned the sting DD$.

Joystick Routine—Line 510 has a USR function to call a machine language routine which allows jumping only when both stick and trigger are pressed. It is POKEd into string STK$ in lines 190-200.

Sound Routines—Two sound routines are used, one with no amplitude change and the other with a decay in amplitude to sound like a piano. Both are inserted in the vertical blank process so as not to interfere with the timing of the BASIC program. They are POKEd into Page 6 using data in lines 1420-1450. The last 10 bytes in line 1450 insert the VBI routine.

Character Set—The character set has been extensively modified for using GR. 4 and GR. 5 graphics. It is first moved to RAM using a machine language moving routine read into ZZ$ in lines 1140-1160. It is moved to the top 1K of memory and RAMTOP is lowered 5 pages in line 1170. Character set data is POKEd in at lines 1180-1300.

Vertical Blank Interrupt—The vertical blank takes care of moving the conveyor belt by modifying the display list. Also the Cricket is also moved when on a belt and tune sounds are updated each VBI.

Display List— A new display list is built in Page 6 on line 1380 using data from lines 1390-1410. Lines 1460-1470 link the display list up to the screen data in DD$ (high order bytes). Lines 1480-1500 provide necessary data to allow the VBI to manipulate the low order bytes of Load Memory Scan instructions in the display list.

This September Game of the Month is being published by **Antic** as a tribute to Stan Ockers—the best-known programmer of public domain games for the Atari. Many newer readers of **Antic** may not be familiar with Ockers. But his smooth, inventive games were a highlight of this magazine's early issues.

The Best Of **Antic** anthology contains two Ockers games, "Chicken" and "Bats." This book is available from the Antic Arcade Catalog in this issue, as are no less than FOUR public domain disks with Ockers games (PD001, PD003, PD004, PD005).

Ockers definitely has a unique way of doing things. He lives in Lockport, Illinois but his games usually make their first appearance in the newsletter of the Eugene, Oregon Atari Computer Enthusiasts club. Ockers resists all offers to turn pro, he insists on keeping all his games (including this one) in public domain and won't accept payment for them.

Therefore, a few readers may have already seen Crickets on bulletin boards here and there. But many more of you are new enough to the Atari so that you will now have the pleasure of experiencing Stan Ockers' bold, clean game programming for the first time.—ANTIC ED

Listing on page 63.

Antic ST Section

September 1985

Resolution: 640 x 400



ST

ST FEVER HITS ENGLAND

"IBM AT Performance"

by CHARLES CHERRY

While on a business trip to Great Britain not long after the Hanover Electronics Fair, I found tremendous excitement over the impending arrival of the Atari 520ST.

ST photos were on the May covers of several major computer magazines and virtually all of the other publications had ST articles. The coverage was uniformly positive.

Practical Computing Magazine, targeted at the business and professional users went to the giant Hanover trade show "prepared to scoff but came away impressed."

WANTED: 130ST

The only negative reactions concerned the dropping of the 130ST. In a home computer market that is still largely cassette-based, the 520ST will cost about twice as much as the average system. Bundled with a disk drive and monochrome monitor, it is priced at 900 British pounds. That's $1,116 at the current exchange rate, or about 25% more than the same ST package in the USA.



Popular Computing Weekly said the 520ST "may still be too expensive to bridge the gap between the home and business in Britain. Atari's decision to drop the 130ST model is a great disappointment."

But not everyone thought the 520ST was too expensive. *Practical Computing* pointed out that "the entire outfit is less than the cost of upgrading a 128K Macintosh". And the *Personal Computer World* reporter, after noting that the cheapest ST system would now cost a lot more money than expected, summed up by saying "Even so, the bottom line is that when the machine appears in the shops, I'll be at the front of the queue to buy one."

The United Kingdom is positive about the software future for the machine too. *Atari User Magazine* reports, "More than 70 UK software companies ordered the GEM Programmers Toolkit on the first day it was available in this country". The article also quotes the UK technical director of Ashton-Tate, publishers of dBase II: "For Ashton-Tate, which is not committed to a sole machine or

continued on page 18

ST ANY WAY YOU WANT IT

Using GEM control panel

by Jack Powell, **Antic** Technical Editor

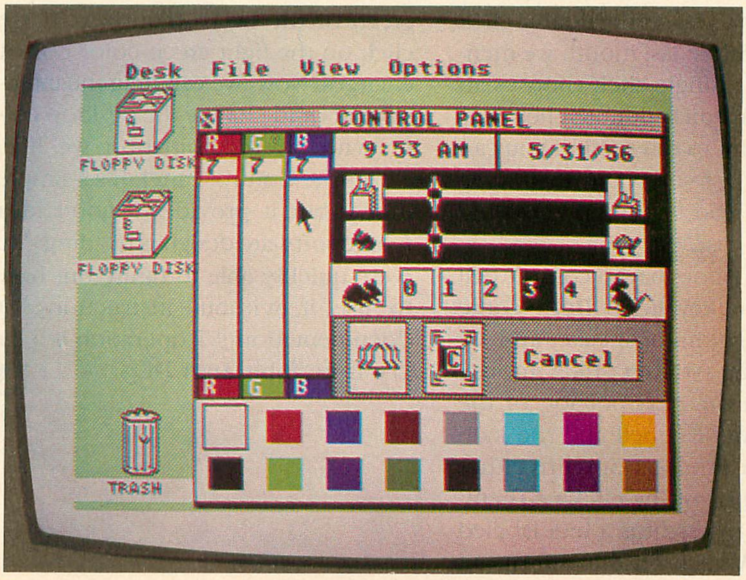


Figure 1

The GEM desktop has a window display called the Control Panel that lets you configure the 520ST to suit yourself. Let's take a look at all that can be done with this useful utility.

When you click the Control Panel option, in low resolution, up pops a colorful window jampacked with icons, switches and buttons (*see Figure 1*).

You'll find that you can click and drag the Control Panel window anywhere on the desktop, or click on the upper-left corner to close it. However, you can't change the size.

REAL-TIME CLOCK

Looking inside the Control Panel window, we see the time is set for 9:50 AM and the date is 5/31/56. This is simply a default set at Atari (probably someone's birthdate). To reset the time, move the mouse cursor to any part of the time and click. A text cursor appears and you enter the current time from the keyboard. The same process sets the date.

Just under the time/date portion are two *slide bars* with an indicator

continued on next page

somewhere along the bar. Left and right of the top bar are icons of a finger pressing a key. By clicking and dragging the indicator along the bar, we adjust the key repeat delay. Key repeat delay is how long it takes for a key to begin repeating after it is pressed.

The next bar down has a tortoise icon on the right and a hare on the left. This adjusts key repeat rate. Repeat rate is how fast the repeats will occur once they begin. It didn't require much guesswork to discover that you move the indicator toward the hare to speed up the repeat rate.

We also found that placing both indicators in the far left position and pressing any key was like turning on a buzzer and made typing impossible. On the slowest setting you could wait a long time before even one character appeared.

Looking down one more section, we find a pair of mouse icons on either side of four numbered squares. Here is where you set the double click rate. If you recall, last month we mentioned that a single click chooses an item, and a quick double-click acts upon an item. In this section you can select how fast you want the double click to be read. Setting one is slow and four is fast.

Yes, the ST does have a console bell and audible keyclicks. Both come through the monitor speaker and are high-pitched—rather like mouse squeaks. In the next portion of the Control Panel is a bell and the icon of a console key. You can turn either of these on or off by clicking them.

Over on the right is a box labeled "Cancel." A click on this returns all Control Panel options to their defaults.

RGB SETTINGS

Now let's play with the colors. In the upper left section of the Control Panel are three vertical columns labeled "R", "G", and "B". They stand for red, green and blue, the three color guns of the RGB video monitor. These three guns may be adjusted to eight levels of intensity in low resolution. Thus, we can obtain 8×8×8=516 colors.

The three RGB slide bars can be dragged up and down the columns to adjust the intensity from zero to seven. Down at the bottom of the Control Panel are 16 boxes representing the ST color registers. One of the boxes is larger than the others to show that it is the register currently affected by the intensity bars *(see Figure 2).*

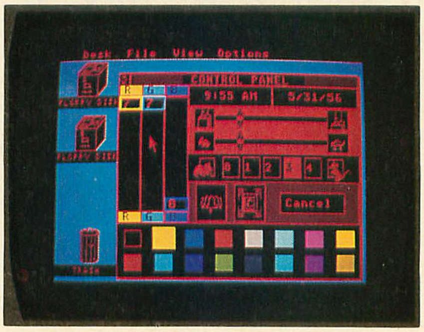


Figure 2

On the desktop three registers are active; black, white, and the light green of the desktop surface. If we click on the light green color box, it enlarges. Now, we can drag the intensity bars and change the desktop color with any combination we like. That's the Control Panel. It's a simple utility, but it provides a good idea where software design is heading. We were quickly able to figure out how to use it without instructions or documentation. ST operation is a far cry from 8-bit Atari DOS.

continued from page 16

system, GEM's easy portability strongly supports our future development strategy."

GEM has also been adopted by ACT, one of Britain's largest computer manufacturers, for its Apricot MS-DOS machine. The Macintosh has not sold well in England, because of its cost. So the Atari may well be the first widespread introduction to what they call "WIMP" (Windows, Icons, Mouse Programs). British software houses are gearing up to ride home on that wave.

MIDI LAN

One hardware feature which is attracting more attention in Europe than in America is the MIDI interface, but not for its musical applications. As *Personal Computer World* says "Even if you don't want to hang a synthesizer onto your Atari, the two MIDI ports needn't be wasted . . . they could make the basis of a very cheap (if slow) local area network." With a transfer rate of 31,250 baud per second, it won't be all that slow.

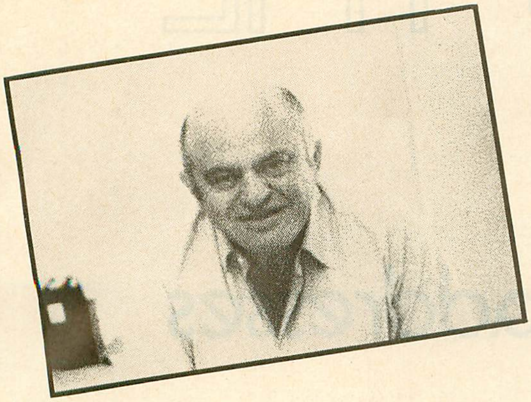
It seems that in Great Britain the 520ST will be thought of as a business computer. While calling it a home machine in the USA, Sam Tramiel told the Europeans that the 520ST offers "performance in the realm of the IBM AT."

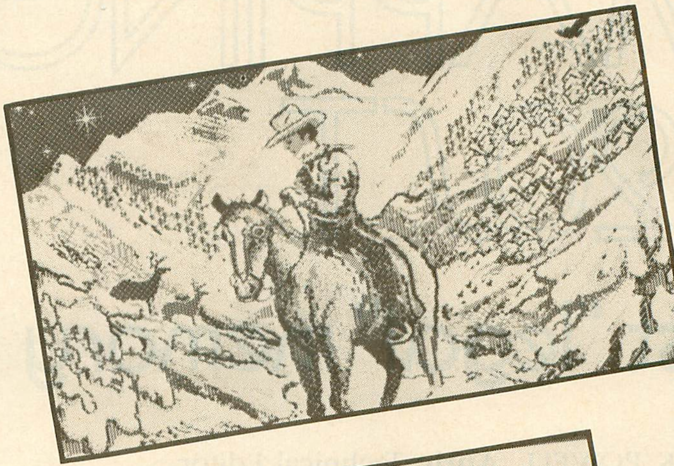
The United Kingdom is a very different computing environment from the United States. It is full of strange machines like Amstrads, Orics, Beebs, Spectrums, and Dragons. The only things an American could recognize are the ubiquitous Commodore 64 and an occasional TI 99/4.

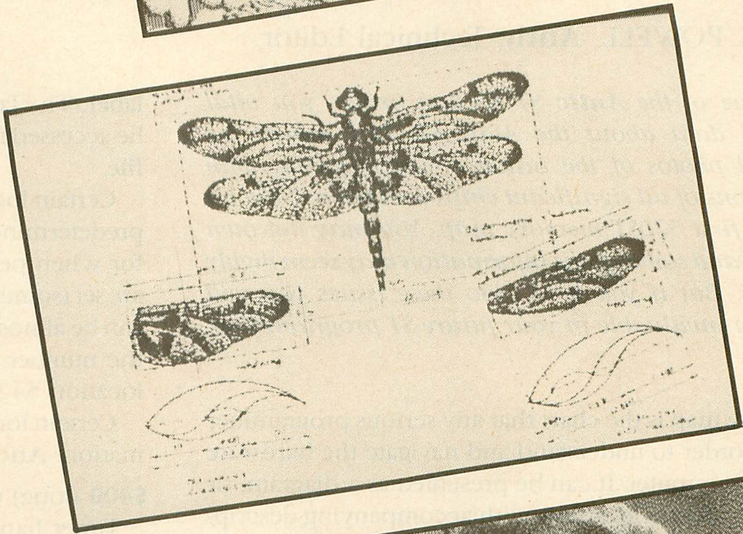
The few Atari owners in England used to gather for passing around battered copies of ANTIC which found their way across the Atlantic. But things are looking up now. The 130XE is selling very fast, software houses are working overtime to make Atari conversions of their titles, and *Atari User Magazine* published its first issue in May. Meanwhile, the arrival of the 520ST is being anxiously awaited by the whole computer community.

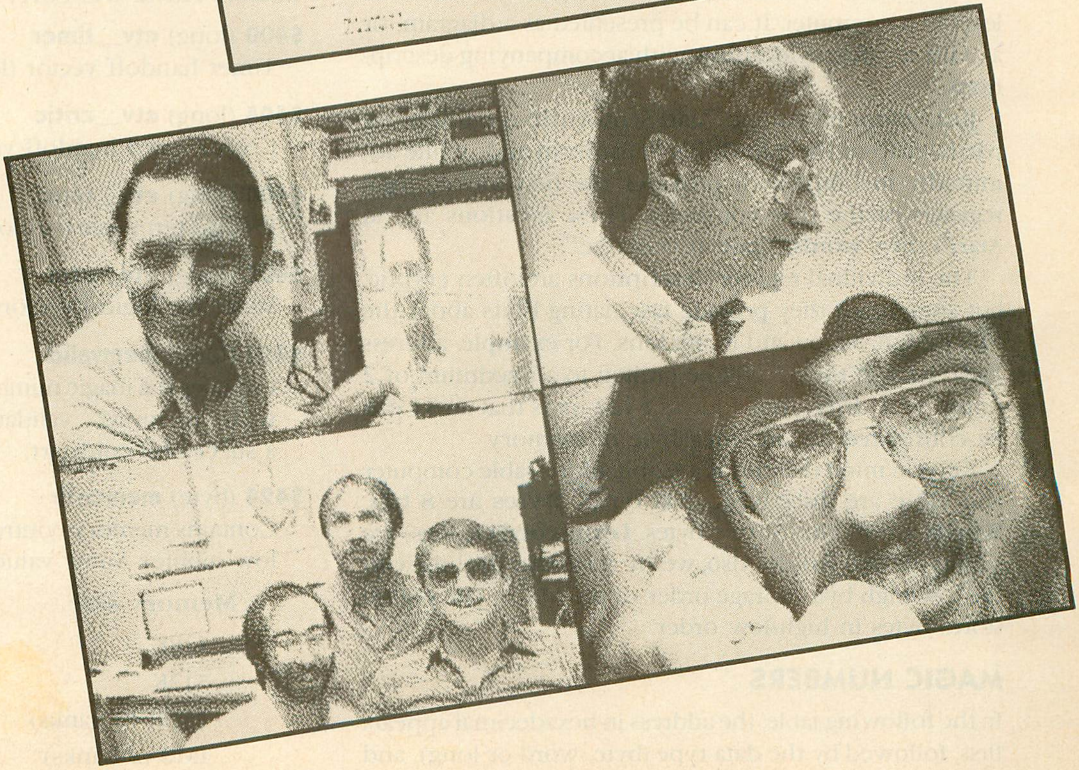
ST GALLERY

At 640 x 400 pixels









Our sleeves were rolled up and we were plumbing the memory depths of our sparkling new 520ST development machine. And then suddenly. . . a package arrived from Atari!

It was our long awaited SM124 high-resolution monochrome monitor—the same one that will be packaged with the first production ST's.

We immediately booted a demo disk we had received from Atari several weeks ago. This disk was packed with the latest ST high-resolution digital pictures—and we'd been unable to use it with our Atari medium-resolution RGB color monitor.

So we were pretty anxious to check out these new graphics, and as you can see from the adjoining reproductions the results are remarkable.

MAPPING THE 520ST

First "legal" memory addresses

by JACK POWELL, **Antic** Technical Editor

Each issue of the **Antic** *ST Section brings you vital technical data about the Atari ST. Last month we presented photos of the inside of the machine, with descriptions of all significant chips and ports. Now we offer the first 520ST memory map. You may not own an ST yet and some of this information may seem highly technical. But if you hold onto these issues you will find them invaluable in your future ST programming.*
—ANTIC ED

A memory map is the chart that any serious programmer needs in order to understand and navigate the hardware level of a computer. It can be presented as a diagram, or as a list of address locations with accompanying descriptions.

Just before press time, Atari sent **Antic** the following partial map of the 520ST. It's in list form and, while admittedly incomplete, it provides the first locked, legal registers of the new computer. These locations are, in Atari's own words, "cast in concrete."

The individual register descriptions are often cryptic, but sometimes they provide fascinating hints about the ST's design, scope and limitations. For example, address $4A6 tells us the ST will be limited to a maximum of 2 floppy disk drives, but address $424 hints that an ST can be configured for one megabyte of memory.

Keep in mind that this is a 32-bit addressable computer and there are three basic data types. **Bytes** are 8 bits. **Words** are 16 bits, or two bytes. **Longwords** (longs) are 32 bits, or two words. Also, we are no longer dealing with the low/high byte storage order of the 6502. The 68000 stores bytes in high/low order.

MAGIC NUMBERS

In the following table, the address in hexadecimal appears first, followed by the data type (byte, word or long), and label. The label is actually a system variable which can be accessed from C when linked with the proper binding file.

Certain locations mention "magic numbers." These are predetermined specific values which the machine looks for when performing certain functions. Magic numbers are set (sometimes whimsically) by Atari programmers and can be almost anything. The computer doesn't care what the number means, as long as it's the right number. At location $426, for example, the magic number is pi.

Certain locations refer to BIOS and GEM specific information. **Antic** will cover this data in future issues.

$400 (long) **etv_timer**
Timer handoff vector (logical vector $100).

$404 (long) **etv_critic**
Critical error handoff vector (logical vector $101).

$408 (long) **etv_term**
Process-terminate handoff vector (logical vector $102).

$40C (long) **etv_xtra**
Space for logical vectors ($103 through $107).

$420 (long) **memvalid**
Contains the magic number $752019F3 which, together with "memval2" validates "memcntlr" and indicates a successful coldstart.

$424 (byte) **memcntlr**
Contains memory controller configuration nibble (the low nibble). Some values include:

Memory size	Value
128K	0
512K	4
256K (2 banks)	0
1MB (2 banks)	5

$426 (long) **resvalid**
If "resvalid" is the magic number, $31415926, on system RESET, the system will jump through "resvector."

$42A (long) **resvector**
System RESET bailout vector, valid if "resvalid" is a magic number. Called early in system initialization (before any hardware registers—including memory controller configuration register—have been touched). A return address will be loaded into A6. Both stack pointers will contain garbage.

$42E (long) **phystop**
Physical top of RAM. Contains a pointer to the first unusable byte ($80000 on a 512K machine).

$432 (long) **_membot**
Bottom of available memory. The "getmpb" BIOS function uses this value as the start of the GEMDOS Transient Program Area (TPA).

$436 (long) **_memtop**
Top of available memory. The "getmpb" BIOS function uses this value as the end of the GEM TPA.

$43A (long) **memval2**
Contains the magic number $237698AA which, together with "memvalid", validates "memcntlr" and indicates a successful coldstart.

$440 (word) **seekrate**

Default floppy seek rate. Bits zero and one contain the default floppy disk seek rate for both drives:

00	6ms
01	12ms
10	2ms
11	3ms (default)

$442 (word) **_timr_ms**
System timer calibration (in ms). Should be $14 (20 decimal) since the timer handoff vector is called at 50hz. Returned by BIOS function "_tickcal" and passed on to the stack to the timer handoff vector.

$444 (word) **_fverify**
Floppy verify flag. When non-zero, all writes to floppies are read-verified. When zero, no write-verifies take place. The default state, after RESET is to verify.

$446 (word) **_bootdev**
Contains the device number the system was booted from.

$448 (word) **palmode**
When non-zero, indicates the system is in PAL mode (50hz video). When zero, indicates the system is in NTSC mode (60hz video).

$44A (byte) **defshiftmd**
Default video resolution. If the system is forced to change from monochrome mode to a color resolution, "defshiftmd" contains the resolution the system will switch to.

$44C (word) **sshiftmd**
Contains shadow for "shiftmd" hardware register.

0	320×200×4 (low resolution)
1	640×200×2 (medium resolution)
2	640×400×1 (high res. monochrome)

$44E (long) **_v_bas_ad**
Pointer to base of screen memory. Always on a 512-byte boundary. Always points to 32K of contiguous memory.

$452 (word) **vblsem**
Semaphore to enforce mutual exclusion in vertical blank interrupt handler. Should be "1" to enable vblank processing.

$454 (word) **nvbls**
Number of longwords that "_vblqueue" points to. On RESET, defaults to 8.

$456 (long) **_vblqueue**
Pointer to a vector of pointers to vblank handlers.

$45A (long) **colorptr**
Pointer to a vector of 16 words to load into the hardware palette registers on the next vblank. If NULL, the palettes are not loaded. "Colorptr" is zeroed after the palettes are loaded.

$45E (long) **screenpt**
Pointer to the base of screen memory, to be set up on the next vblank. If NULL, the screen base is not changed.

$462 (long) **_vbclock**
Count of vertical blank interrupts.

$466 (long) **_frclock**
Count of vertical blank interrupts that were processed (not blocked by "vblsem").

$46A (long) **hdv_init**
Vector to hard disk initialization. NULL if unused.

$46E (long) **hdv_dsb**
Vector to routine to return a hard disk's state block. The WORD device number should be on the stack. NULL if unused

$472 (long) **hdv_bpb**
Vector to routine to return a hard disk's BIOS parameter Block (BPB). Same calling conventions as the BIOS function for GETBPB. NULL if unused.

$476 (long) **hdv_rw**
Vector to routine to read or write on a hard disk. Same calling conventions as the BIOS function for RWABS. NULL if unused.

continued on next page

$47A (long) **hdv__boot**
Vector to routine to boot from a hard disk. NULL if unused.

$47E (long) **hdv__mediach**
Vector to routine to return a hard disk's media change mode. Same as BIOS binding for floppies. NULL if unused.

$482 (word) **__cmdload**
When nonzero, an attempt is made to load and execute COMMAND.PRG from the boot disk. (Load a shell or application in place of the desktop.) Can be set to nonzero by a boot sector.

$484 (byte) **conterm**
Contains attribute bits for the console system:

Bit	Function
0	non-zero: enable keyclick
1	non-zero: enable key-repeat
2	non-zero: enable bell on ^G

$48E (long) **themd**
MD for GEMDOS. Can be manipulated by boot sectors.

$49E (word) **________md**
More MD for GEMDOS. Can be manipulated by boot sectors.

$4A2 (long) **savptr**
Pointer to register save area for BIOS functions.

$4A6 (word) **__nflops**
Number of floppy disks attached (0, 1, or 2).

$4B4 (long) **__bufl**
Two buffer-list pointers.

$4BC (long) **__hz__200**
Raw 200hz system timer tick. Used to divide by four for a 50hz system timer.

$4C4 (long) **__drvbits**
32-bit vector, returned by the "DRIVEMAP" BIOS function, of "live" block devices. If any floppies are attached, this value is 3.

MORE ST PRODUCTS COMING

Two new ST models, 500 megabyte CD ROM, plus lots of software in '85

by MIKE CIRAOLO and JACK POWELL

At the Consumer Electronics Show in June, Atari announced that two new ST models and a pioneering 500 megabyte ROM compact disk system would appear on dealers' shelves in time for the Christmas rush.

Complete lines of integrated software for the ST were announced by three major developers. First programs from the emerging productivity series were to ship this autumn from Haba, Batteries Included and Rising Star. Haba also promised ST owners a 10 megabyte hard disk for $499 and a $299 Hayes-compatible modem.

NEW ST MODELS

Atari said that both of the new ST's will have 256K RAM for programmable memory. The 260ST will retail at $399. The 260STD is to be $499

and includes a built-in 3.5" disk drive. Otherwise they are identical with the 520ST except for the following:

•GEM and the rest of the TOS operating software will be on ROM chips instead of on disk.

•The television RF modulator is to be built-in.

•The new 256K models won't ship till October or November.

How is this significantly different from the 520ST?

According to Atari Marketing Vice President James Copland, the first 2,000 U.S. units of the 520ST were shipping in June to Atari users groups. In July the 520ST would appear in computer specialty stores, and mass merchandiser distribution of the ST line would begin in the fall, Copland stated.

By June, the 520ST was already on computer store shelves in Canada and parts of Western Europe.

Price of the 520ST was set at $799 and included a 3.5" disk drive, a high-resolution monochrome monitor, an external RF modulator pack—and GEM on disk, leaving 256K RAM as free memory after loading GEM and TOS.

So all the 1985 ST's will now have 256K of usable RAM. . .or will they?

ROM OR NOT?

For pre-Christmas delivery, manufacturing must begin no later than September. The CES announcement of the 260ST and 260STD gave Atari all of June and July to make sure GEM and TOS were thoroughly debugged.

These newly announced 260 models will allow Atari to maintain credibility by meeting its pledge to ship the 520ST to US stores in early July. Disk updates could easily remedy any bugs found in the operating system of early 520ST's manufactured in May and June.

To **Antic**, the whole thing looks like another gutsy, innovative move from Atari Chairman Jack Tramiel. Much of the U.S. business press unfortunately is computer-illiterate and reports even minor production delays as putting a company's entire future in doubt. Atari *needed* to bring an ST to market as quickly as possible, even if in limited numbers.

Yet Tramiel had clearly learned a valuable lesson during his Commodore days, with the glitchy Commodore 64 operating system that went into ROM sooner than it should. In the long run, rushing GEM and the ST operating sytem into chips before it was truly ready would create user problems and be bad business.

Before and during CES, Atari kept saying that the 520ST model would never have GEM and TOS in ROM.

But just as this issue went to press, Atari president Sam Tramiel issued a statement, confirmed by **Antic**, that the entire finalized 520ST operating software would be made available on simple plug-in chips at "nominal cost."

CD ROM

The mind-boggling 500 megabyte CD ROM was the hottest thing at CES. It was displayed at the Atari booth by Activenture, which is developing the technology under contract with Atari.

Imagine a read-only disk that's identical to a compact audio digital disk, but which could contain a 100 volumes of reference books with room left over. And the 100 volumes of information would be instantly accessible. Only three seconds were required for the 520ST and CD ROM to search a keyword through an entire 26-volume encyclopedia.

CD ROM will work with any material that can be digitally encoded—video images, software, photographs, etc.

An exclusive interview next month in the October, 1985 **Antic** will cover the CD ROM breakthrough in depth.

NEW SOFTWARE

Software developers, whether at CES or responding to **Antic's** monthly survey, showed increasingly enthusiastic support for the ST computers.

Batteries Included announced a major ST commitment with their IS integrated software based on the GEM icon/window/mouse environment. According to Michael Reichmann, director of product development, the entire IS line will work at an intuitive level so that users won't need to memorize any commands.

The IS line will include a word processor with built-in spelling checker, a combined spreadsheet and graphics package, a database manager, and a stock portfolio manager. Screen structure and layout will be the same for all programs.

All programs in the IS series will be released for the Atari ST and for the IBM PC and compatible market. The first program, Portfolio, is to be available for the ST in September. Famed stock analyst, Lee Isgur is the designer.

HABA HIPPO

Haba Systems, of Van Nuys, California announced a September multiple release for the ST. There will be a word processor, HabaWord; a file and report manager, HabaFiles; a spreadsheet and graphics package, HabaCalc 'n' Graph; and a communications program, HabaCom.

The company, which released the integrated /// Easy Pieces spreadsheet, word processor and database for the Apple ///, also said it would bring out their Hippo C language and a checkbook program for the ST in late July or early August.

Haba is known for having released over 10 Macintosh products. Now the company will convert all of its current and future programs from the Mac to the Jackintosh, starting in January 1986.

In addition to its line of software, Haba told **Antic** they will sell a 10 megabyte ST hard disk for $499 and a fully Hayes-compatible modem for $299. These products are called HabaDisk and HabaModem.

ST VALDOCS

The highly praised Valdocs application software series, which had been previously implemented on the Epson QX-10 computer, will be released through Atari by Rising Star Industries. The software may be retitled, and the first two of 14 integrated modules were to come out this summer.

continued on next page

Among the 14 modules are a spreadsheet, database, word processor, paint program and more. The first two modules are expected to be a telecommunications program and a CAD/CAM system. Later Rising Star is to release a "core" module which integrates the entire series.

AND GAMES

Sierra On-Line will convert the submarine simulation game GATO for the ST. GATO puts the player in the role of a World War II submarine commander, searching out and attempting to destroy the Japanese Imperial Fleet.

Sierra president Ken Williams said "We want to see exactly how large these two new markets are. We figure the best way to test unknown waters is to publish a proven product with broad appeal." GATO is already out for the IBM PC, Macintosh and Apple II with 128K.

SubLogic, publisher of the best-selling Flight Simulator II will release a new simulator called Jet. Electronic Arts is expected to convert their Financial Cookbook for the ST.

Accolade/F.T.L. Software, producers of the Apple II game Sundog: Frozen Legacy are converting the prize-winning hybrid game, a role-playing economic space adventure.

Several 8-bit computer games are being adapted for the 16-bit ST. Rugby Circle is working on the arcade hit Joust, for release by Atari. Datasoft plans on Zorro, a sort of of Bruce Lee with swords, and Goonies, based on the summer Spielberg movie.

FORTH & WINNER

Forth language fans can look forward to a massive 83 Standard Forth system from the Dragon Group. It will come in three packages which include the basic Forth implementation without GEM, a Forth system including GEM calls, and an optimizer package which is intended to dramatically improve the speed of your final code. The company says it has clocked their product to be twice as fast as Mac Forth.

And Abacus Software gets the prize for delivering the very first completed ST product to **Antic's** office. We received their book, "Presenting the Atari ST" just as we were going to press and will review it next month.

ST DEVELOPERS

Atari Corp.
1196 Borregas Avenue
Sunnyvale, CA 94086
(408) 745-2000

Haba Systems
15154 Stagg Street
Van Nuys, CA 91405
(818) 901-8828

Batteries Included
30 Mural Street
Richmond Hill, Ontario
L4B 1B5 Canada
(416) 881-9941

Rising Star Industries
25500 Hawthorne, Suite 2000
Torrance, CA 90505
(213) 373-9112

Rugby Circle
1251 Rugby Circle
Bloomfield Hills, MI 48013
(313) 362-0860

Dragon Group
148 Poca Fork Road
Elkview, WV
(304) 965-5517

Abacus Software
P.O. Box 7211
Grand Rapids, MI 49510
(616) 241-5510

Datasoft
9421 Winnetka Avenue
Chatsworth, CA 91311
(213) 701-5161

Electronic Arts
2755 Campus Drive
San Mateo, CA 94403
(415) 571-7171

Sublogic Communications Corp.
713 Edgebrook Drive
Champaign, IL 61820
(217) 359-8482

Accolade/F.T.L. Software
20863 Stevens Creek Boulevard
Cupertino, CA
(408) 446-5757

Activenture
2511-C Garden Road
Monterey, CA 93940
(408) 375-2638

Sierra On-Line
Sierra On-Line Building
Coarsegold, CA 93614
(209) 683-6858

REVISION C CONVERTER

Type-in fix for buggy BASIC revision B

by MATTHEW RATCLIFF

Save $15 (plus $2.50 for shipping) as well as a whole lot of time and aggravation. This type-in Autorun file converts into Revision C BASIC the buggy Revision B that was built into most Atari XL's. Requires 48K, a disk drive and, of course, an Atari XL computer with Revision B BASIC.

*Antic has done it again. Our Antic Arcade Catalog **Fix XL** disk ($10, PD026) provided an improved Atari Translator Disk so you could run early software on your XL models. Now here's a type-in Revision C Converter. It's just what you've been waiting for if you own an Atari 800XL (or 600XL expanded to 64K) and want to convert your fatally buggy Revision B BASIC into Revision C—but haven't been able to get the Revision C cartridge from Atari.*—ANTIC ED

Most 600XL and 800XL computers have a defective BASIC called Revision B (Rev. B) in ROM. And most people don't know that **EVERY time you save a file with Rev. B, 16 useless bytes are added to that file.** If you LOAD and SAVE the file enough times, you could run out of memory even though you never added a single line of code.

The Rev. B bug can be avoided only if you use the LIST and ENTER commands instead of SAVE and LOAD. The problem with this is that another bug often causes complete system lock-up when attempting to ENTER large files.

BUGGY HISTORY

Atari has gone through three versions (or Revisions) of BASIC. The original Atari BASIC, Rev. A, was written for Atari 400/800 computers by Optimized Systems Software (OSS). It was one of the most powerful 8K BASICs ever produced.

Unfortunately, there was a fairly nasty bug in Rev. A. Occasionally, the computer would lock while editing a program.

This syndrome became affectionately known as "keyboard lockup" since the computer would not recover, even if [RESET] was pressed. The only way to "recover" was by turning off the machine and and losing everything you had been working on.

The problem resulted from the fact that this version of Atari BASIC did not handle the "carry bit" properly. The lockup occurred if you deleted lines of code in such a way that a multiple of 256 bytes of "BASIC tokens" were removed from your program.

Fortunately, this did not happen often. Out of necessity, many Atari owners soon learned the good habit of saving programs frequently. If a lockup occurred, you would only lose changes which had been made since the last SAVE.

When Atari began producing the newer 600/800XL computers, they decided to fix this bug by writing their own new Rev. B BASIC. Unfortunately, in the process the much more

continued on next page

"Every time you save a file with Rev. B, 16 useless bytes are added . . ."

serious 16-bits-added bug was introduced.

In order to determine exactly which revision of Atari BASIC you own, you can perform the following test:

PRINT PEEK(43234)

VALUE	VERSION
162	A
96	B
234	C

The Rev. B bug was first documented by **Antic** in "Exploring The XL," June, 1984. Although the old Atari knew about this bug even before the **Antic** article was printed, they continued to produce Rev. B computers into early 1985.

Why? Apparently they already had purchased enough Rev. B ROM's for the 1984 production year. ROM chips are usually purchased in large quantities, to achieve lower cost per unit. Atari, Inc. decided that the bug was not serious enough to warrant discarding all the defective Rev. B ROM's. This was unfortunate, to say the least . . .

MORE HEADACHES

Those 16 extra bytes bestowed upon your files by Rev. B can cause many other problems besides just gobbling up memory.

Many people run into Error 9, String Not DIMensioned, at the very program line where the DIM occurs. The [RESET] key, the CLR command, even LISTing and ENTERing the file cannot cure the Error 9 problem. The program becomes useless.

Most often, with fairly large files of 16K or more, keyboard lockup occurs. This happens when editing a program or just by LOADing a file that has been "SAVEd one time too many." How many times is too many? That depends on your original program.

LISTing and ENTERing files circumvents the "disappearing memory" bug, but then the Error 9 problem seems to occur much more frequently. *(Our in-house experience has more often shown computer lock-up when ENTERing files of a particular size. This can be fixed by adding or deleting one or more bytes of the file—IF you can get to it!*—ANTIC ED)

REVISION C

Atari BASIC Rev. C has been available since June, 1984. But it only began appearing in XL computers in late March 1985. At this writing, there are still many Rev. B computers being sold off the shelf.

It is now possible to obtain Rev. C on cartridge directly from Atari. You can send $15 (plus $2.50 for shipping) to:

Atari Corp.
Customer Relations Dept.
1196 Borregas Avenue
Sunnyvale, CA 94086

However, many **Antic** readers have found that delivery of this product is slow at best—5-week waits are not uncommon—and can be maddeningly inconsistent at times.

You also have the option of purchasing one of the excellent third party BASICS available for the Atari. **Antic** has been particularly impressed with **BASIC XL** ($79, Optimized Systems Software). Most of the independently made BASICs cost more than $15, but they also offer greater power than any BASIC yet released by Atari itself, including Rev. C.

Or if you can find a Rev. A cartridge, you can plug it into your XL and it will take over from the built-in Rev. B.

But now, for the first time you have a fourth choice. If you want Atari Rev. C BASIC right NOW and for FREE, here's how to get it . . .

REV. C CONVERTER

I wrote Rev. C Converter to give people a do-it-yourself debugger of Rev. B.

Listing 1 is a BASIC AUTORUN.SYS file creator. Type it in, check it with TYPO II, and SAVE or LIST a copy to disk. Before RUNning it, place a formatted, **DOS 2** disk in your drive.

Antic Disk subscribers will find Listing 1 under the filename REVB2C.BAS. Also on the disk is the binary file REVB2C.EXE which can be transferred to another DOS 2 disk and renamed AUTORUN.SYS for greater convenience.

The program reads data and creates a binary file on your DOS disk. Whenever you boot up your 800XL (or 600XL with 64K), this file loads into Page 6 memory (locations 1536 to 1791) and copies your bugged Rev. B BASIC to its "shadow RAM." Once installed, Page 6 is available for your use, as long as you don't press [RESET]. If you leave Page 6 alone [RESET] will not disable your new Rev. C.

USING DISKIO

If you wish to use DISKIO (**Antic**, January 1984) with this program, you can append the Converter binary file to the DISKIO binary file as follows:

First, rename your Converter AUTORUN.SYS file to REVB2C.EXE, then copy it to a disk with the DISKIO AUTORUN.SYS file on it. Now, from the DOS 2 menu, append the files by typing [C] [RETURN] REVB2C.EXE, AUTORUN.SYS/A [RETURN]. Don't forget the /A.

ONLY 12 BYTES

Listing 2, REVB2C.M65, provides the MAC/65 source code for more advanced programmers. You may wish to study it to see how the ROM/RAM bank switching is done and what the actual changes are in Rev. C.

Remarkably, there are only 12 bytes different between Rev. B and Rev. C, but they make ALL the difference!

Here is a simple before-and-after program to demonstrate the problem with Rev. B and how it is fixed in Rev. C:

```
10 ? FRE(0):SAVE "D:JUNK"
:RUN "D:JUNK"
```

continued on page 28

TIC TOC FLIP

Eye-popping demonstration

by GENE LEVINE

This short BASIC program is the best demonstration of the page flipping animation technique that we've seen. It will run on all Atari computers with 32K disk or 24K cassette.

*Page flipping is a technique in which two or more screen pictures are drawn in memory and then pointed to by altering the display address of the display list. The accompanying article assumes a degree of familiarity with the concept of page flipping. For more information on this subject, refer to "Page Flipping" by David Plotkin (**Antic**, January 1984).*—ANTIC ED

It's easier to think of page flipping as screen flipping, because you are not really flipping a page at a time. Actually you are flipping the multiple of pages needed to make a screen of the particular graphics mode you are using.

A page is a 256-byte chunk of memory. Four pages would thus be 1024 bytes or 1K. The number of pages used for a particular screen display depends on which graphic mode you choose (See the Table at the end of this article.) Tic Toc Flip uses 11 Graphics 5+16 screens and requires 8 pages for each screen display.

MEMORY MANAGEMENT

The more memory a graphics mode requires, the more pages it needs, and that means less screens are available to flip. More RAM means more pages available, so those users with less than 48K will be very limited in flipping applications. In fact, even with 64K, you'll find the high resolution modes of little use for this kind of animation.

Type in Listing 1, check it with TYPO II and SAVE a copy before running it.

When RUN, the program will draw a series of nested boxes. After drawing each screen in a different color, Tic Toc Flip will animate the screens into a tipsy pyramid with a "tick tock" sound.

You can speed up the animation by pressing [+], and slow it down by pressing [-]. To end the program, press [ESC] which will reset the initial values of locations 106 and 561. If these are not reset, the program will have little if any memory left to use.

PROGRAM ANALYSIS

The DISPLAY LIST is a set of instructions the computer uses to display data to the screen. Since the display data can be stored in free areas of memory, a screen can be recalled instantly.

This is accomplished in the subroutine beginning at line 100 which subtracts multiples of 8 pages (variable TX) from RAMTOP and then sets up another Graphics 5+16 screen.

Lines 200 through 310 do the plotting and drawing, incrementing the horizontal and vertical variables with

continued on next page

"... animate screens into a tipsy pyramid with a "tick tock" sound."

each pass, before RETURNing. A variable is made of PEEK(561) for the first and last screens so that the pages may be added or subtracted for back and forth movement.

Line 1500 sets colors 1–3 and will turn the screen on. The screen may be turned off at line 100 by POKEing 559 with a 0 instead of 34. This will speed up the drawing process somewhat.

Lines 20 to 30 are the animation loop. The FOR NEXT loops increment and deincrement PEEK(561) by dividing 88 (11 screens times the 8 pages required for a GR. 5+16 screen) by 8 and adding or subtracting that value from PEEK(561) for the back and forth movement.

Line 40 is GOSUBed with each screen call to set the desired speed. Try removing the GOSUB 40 commands from lines 20 and 25 if you want to see the real speed of flipping.

Line 2000 initializes the following variables:
DH - PEEK(561) default of 156
T - PEEK(106) default of 160
TX - the amount subtracted from T
V & VV - vertical offset variables
X & XX - horizontal offset variables
S - speed variable

The program has been structured for speed of execution. The closer to the top of a program, the faster a loop or GOSUB will execute. This is why the animation loops are at the very top. The screen draw routine is next so that it may draw as fast as possible.

PAGES PER MODE TABLE

The following table was prepared with 32271 bytes of user memory. The results reflect the absolute maximum of screens available for flipping that I could squeeze into a bare bones program. Results will not only differ with more or less RAM but with more or less programming code.

GR. MODE	Number of Screens	Number of Pages
0	31	4
1	31	4
2	62	2
3	62	2
5	15	8
7	4	32
8–11, 15	3	32

Gene Levine is the author of "Hypnosis with Brainwave Synchronization" which is marketed by XLENT software. A former artist, he is currently employed as a house painter.

Listing on page 62

REVISION C CONVERTER
continued from page 26

RUN this on BASIC Rev. B and you can watch your memory disappear 16 bytes at a time. If you allow this program to RUN long enough, you will get an Out Of Memory Error and maybe computer lockup too. You could also get "scrambled disk" and lose all your disk files—**so be sure to use only a "junk" disk with NO files you care about losing.**

Now re-start your system with the Rev. C autorun file and try the sample above. Your computer will no longer gobble up memory.

You might still consider purchasing a Rev. C cartridge since this program gives you a RAM based BASIC that is not as crash proof as ROM cartridge programs. The program does check for an "external" cartridge and takes no action if one is plugged in. This way it won't crash your non-BASIC cartridges.

RAMTOP FIX

If you have problems with programs crashing, try moving RAMTOP down by 1K for graphics modes 0-6 and by 4K for modes 7-11. It is simple to control RAMTOP:

```
10 POKE 106,PEEK(740)-4:R
EM MOVE RAMTOP DOWN 1K, 4
 PAGES
20 GRAPHICS 0:REM RESET T
HE DISPLAY LIST
```

Use PEEK(740)-16 for 4K. I have run several programs in this RAM BASIC with high resolution graphics and P/M graphics without having to move RAMTOP. Ths OS in the old 400/800 systems would sometimes write above RAMTOP, when executing certain screen control functions. This is supposed to have been fixed in the newer XL computers. If you should have a problem with BASIC programs crashing with the RAM Rev. C, moving down RAMTOP might cure it.

*Matthew Ratcliff has performed yet another service for **Antic** readers by writing this fix for Atari BASIC Rev. B. We are eagerly awaiting reader entries in our contest based on Ratcliff's ATARI 'Toons (August, 1985).*

Listing on page 54

ATARI TIME MACHINE

Machine language timing power

by FRED PINHO

Timer displays are important in many different kinds of programs. For example, game designers couldn't live without timers. Now this article teaches you how to harness the power of Atari's timer registers for your own programs. (To get the most out of this information, you should understand the essentials of machine language programming or be an intermediate BASIC programmer.) Demonstration listings are given in both machine language and BASIC. They work on all Atari computers of any memory size, with disk or cassette.

This tutorial explores the electronic gears of the Atari timepiece. Many types of programs are enhanced by timer displays. The Atari has several timer registers which can be programmed in BASIC, but the most accurate timers are written in machine language.

REAL-TIME CLOCK

BASIC programmers are probably most familiar with the real-time clock. This is a three-byte count-up timer at locations 18-20 ($12-$14) in memory. The least significant of the three bytes is stored in location 20. Every sixtieth of a second (called a "jiffy") the Atari Operating System (OS) increments this byte. When the value "overflows" (counts past 255) it is reset to zero and the value in location 19 is increased by 1.

Similarly, when the value in 19 overflows, it returns to zero and the value in 18 is incremented. The value in 18 overflows approximately once every 78 hours and all three registers are returned to zero. The three-byte, real-time clock timer can be represented by the following table:

Memory location	18	19	20
Seconds per count	1094	4.272	0.0167

To use this timer in your BASIC program, you must first calculate the total number of jiffies to be timed:

```
JIFFIES=PEEK(20)+ PEEK(19)*256+
PEEK(18)*65536
```

Now, convert jiffies into minutes and seconds:

```
MINUTES=INT(JIFFIES/3600)

SECONDS=((JIFFIES/3600)-MINUTES)
* 60
```

The biggest problem with using these registers in a BASIC program is that complex BASIC programs, such as games, tend to slow down BASIC timers. The slowdown can be avoided by using machine language timers.

COUNTDOWN TIMERS

The OS maintains five two-byte countdown timers which count backward from a given number to zero at an approximate rate of 60 times per second. When the timer reaches zero, either a flag byte is set, or a user-defined machine language subroutine is executed. These timers are explained in *Figure 1*.

You must know machine language for timers one or two because the subroutines they call must be written in machine language. The remaining timers may be used from BASIC, but will have the same limitations as the real-time clock.

Figure 1

Count Down Timer Number	Located at	Action on Reaching Zero	Address of Subroutine Stored at	Flag Byte Located at	Comments
1	536,537	Calls subroutine	550,551	—	Reserved for OS
2	538,539	Calls Subroutine	552,553	—	
3	540,541	Zeros flag byte	—	554	Used by cassette
4	542,543	Zeros flag byte	—	556	
5	544,545	Zeros flag byte	—	558	

VERTICAL BLANK

The timing routines we will demonstrate use the vertical blank interrupt (VBI) which occurs approximately every sixtieth of a second while the video electron gun is turned off and re-aimed at the top of the screen. Since the VBI actually occurs 59.92334 times per second, our calculations will contain an error of 0.13 percent, or one-third of a second every 255 seconds. This is still accurate enough for our purposes.

THE PROGRAMS

Listings 1 and 3 are BASIC programs which demonstrate timers through the use of assembly language routines. The source code for these routines is in Listings 2 and 4. Listing 5 is an example of an all-BASIC timer for comparison.

Type in the BASIC listings, checking them with TYPO II and SAVEing copies. The assembly listings are for instruction and do not need to be typed.

AL COUNTDOWNS

To properly enable the countdown timers, the Atari OS has a machine language routine called SETVBV located in ROM at 58460 ($E45C). To use SETVBV, first store the address of your machine language subroutine (for timers one and two) or set the flag byte to a positive value (for timers three, four, and five). Next, store the countdown time, in jiffies, in the X (high-byte) and Y (low-byte) registers. Finally, store the timer number in the accumulator and code a JSR SETVBV.

BASIC COUNTDOWNS

First, POKE a 1 into memory location 66 ($42). This allows a routine called CRITIC to temporarily suspend the updating of timers and various other functions. Next, load the countdown time (in jiffies) into your timer routines and set the flag byte. Finally, reset CRITIC to zero. To keep your timers accurate, you must always reset CRITIC to zero as quickly as possible.

LISTING 1

Lines 120-220 are the machine language routine which is stored in Page Six of memory. Also used are seven bytes of the cassette buffer (1021-1026, 1028) and two Page Zero bytes (208,209). These bytes function as shown in *Figure 2*.

The BASIC program first asks for the initial value of the countdown timer, which can be up to 250 seconds. The total number of jiffies is calculated at line 40. This value is changed into low-byte/high-byte format and POKEd into locations 1025 and 1026. The timer active flag is also set to zero.

When the program is RUN, the time will be displayed in the lower right corner of the text window. At lines 50 and 60, the position of the text window is determined and the address of the MINUTES display is POKEd into locations 208-209. The timer routine automatically enables timer five in line 70. Here, the timer routine is linked to the vertical blank interrupt.

Finally, location 558, the flag byte, is monitored for a zero value in line 90. This is the signal that timer five has run out. Before checking the flag byte, a short delay is used in line 80 to allow the OS to set up and enable the timer.

LISTING 3

Listing 3 uses the real-time clock at locations 18-20. In this case, the lowest two bytes (19, 20) are used, which limits the timer to 4 minutes, 15 seconds.

Again, the routine is stored in Page Six. Location 1027 holds the task done flag. Initially set to zero, BASIC

continued on page 34

Figure 2

Memory Locations	Function
208-209	Contains the desired starting location of the timer display in lo-byte/hi-byte format. Set by BASIC.
1021	Timer active flag. BASIC initializes it to zero to indicate that the timer must be enabled. Enabling is done automatically by the routine.
1022-1024	Used internally by the timer routine.
1025-1026	BASIC stores the total countdown time (in jiffies) here. The value is stored in lo-byte/hi-byte format.
1028	A delay counter. Used internally by the timer routine.

MIRRORED DISPLAY LISTS

Fast, unusual graphics in half the memory!

by DAVID PLOTKIN

Throw splashy kaleidoscopic patterns all over your screen. Learn how to use mirrored display lists—a powerful graphics programming technique that easily produces distinctive effects without tying up a great deal of memory. The short BASIC demonstration program works on all Atari 8-bit computers of any memory size, with disk or cassette.

They say you can't teach an old dog new tricks, but there still seems to be no limit to the tricks you can teach your Atari. This article demonstrates an interesting screen modification that can lend an unusual and impressive look to your programs with relatively little effort.

REFLECTION DISPLAYS

Inside your Atari is a custom microchip called ANTIC that controls the computer's video output (Yes, that's where the name of your favorite magazine came from.) Among the things that ANTIC does is set up the screen display and track which memory locations hold the required data for display.

This chip gets its instructions from a program called the display list, which is written into RAM by the 6502 microprocessor every time a GR. command is made.

If you modify a display list so that the bottom half of the screen 'reflects' the top half, then anything drawn on the top half also appears upside-down on the bottom half. One good use for this effect would be to depict stars emerging from the center of the screen and moving toward the edges for a three-dimensional effect.

Your program only needs to be concerned with drawing and moving the stars in the top half of the screen, since those on the bottom are their mirror image. This gives you fast-moving effects because you're only manipulating half the normal amount of data to create graphics.

And since you're using the same memory area for both halves of the screen, you save half of the normal memory requirement for a graphics mode. In Graphics 8, that's about 4,000 bytes—a substantial amount!

HOW IT WORKS

Type in Listing 1, checking it with TYPO II. SAVE a copy before you RUN it. **Antic** disk subscribers will find the program under the filename REFLECT.BAS.

The best way to understand how to use the reflection technique is to roll up your sleeves and get to work.

In line 1000, we find the highest available memory address, then step back 4K for the screen (YSTART), and another 1/2K for the display list (DLS). Line 1010 puts the address of the display list into the location that tells BASIC where the top of usable memory is.

Then we start POKEing a display list into memory. If you're familiar with display lists, you'll note that it looks like a Graphics 8 setup,but with only half the normal number of scan lines—96. Next, we calculate the address of the middle of the screen by

continued on next page

"...depict stars emerging from the center of the screen and moving toward the edges..."

multiplying the number of lines (96) by bytes per line (40) and adding this to YSTART.

Lines 1040 to 1060 contain the secret of our trick with the display list. In each display list instruction for the bottom half of the screen, the LMB (Load Memory Scan) option is set, telling the computer to treat the next two bytes as the address from which to display data for that line.

This address, which we must POKE in for each line of the bottom half of the display, starts at the middle of the screen (as computed in line 1025), and is decreased by one line (40 bytes) for each remaining line of the display list. Therefore, the bottom line of the screen displays data from the same address (the same data, in other words) as the first line of the screen, and the bottom half of the screen mirrors the top.

Lastly, line 1070 ends the display list by inserting a JMP instruction (65) and telling the computer where the beginning of the list is. Line 1080 tells the computer to use the new display list by putting its address in memory locations 560 and 561. The remaining lines create pictures to demonstrate the mirrored screen.

*Dave Plotkin of Walnut Creek, California is a regular contributor to **Antic** in both BASIC and ACTION!*

Listing on page 62

"Many programs are enhanced by timer displays."

ATARI TIME MACHINE
continued from page 31

stores a positive number here when the task is completed, signaling the timer routine which is monitoring this location for a positive number.

A modified display list is used to insert a Graphics 2 text line between the Graphics 7 lines and the text window. Calculations in lines 30 and 40 place the timer display in the middle of the Graphics 2 line. By storing the starting address of the timer display in locations 208-209, the programmer can point the display to any text line on the display list.

The program is designed to monitor the time required to plot the function R=COS(4*Q) in Graphics 7. This is to demonstrate that BASIC is not significantly slowed by machine language timers. Listing 5 lets BASIC perform all of the timer functions without the help of machine language. RUN the two programs and compare results. Here are some sample results:

	Time In Seconds
No Timer	168
Machine Language Timer	168
BASIC Timer	218

As you can see, the machine language timer ran significantly faster, and ran less than a second slower than the same program without a timer.

DELAYED DISPLAY

Although these routines are running every sixtieth of a second during the VBI, to increase efficiency the calculation and printing of time is done only every sixth of a second. Since the minimum time increment displayed is one second, there is no need to update the display more frequently.

A delay counter at location 1028 is used to track our display update. A 10 is initially stored there and every VBI decreases the contents of this location by one. The time is calculated and printed only when the delay counter reaches zero.

To change the printing frequency, replace the 10 with your own number in line 160 of listing 1 and line 140 in listing 3.

*Fred Pinho of Valley Cottage, New York is a biochemical research engineer, a part-time tennis pro and a regular contributor to **Antic**.*

Listing on page 59

COMPUTER PALACE WE KNOW ATARI!

CALL US AT (503)683-5361 FOR INFORMATION

Help Calc

SPREADSHEET TEMPLATES READY TO LOAD-N-GO FOR USE WITH *SynCalc*

"Software fit for a King!"

Royal Software

Let Help Calc Help You For Only . . .

$19.95

FEATURES:

- Eight ready-to-use Templates.
- Easy to use . . . Just load them from Syncalc and enter your data.
- Save hours of tedious spreadsheet setup.
- All formats are pre-tested for accuracy.
- Complete, professionally written instruction manual with spreadsheet examples.

Requirements:
Syncalc program. 48K (min) Atari computer. At least one disk drive. 80 column printer.

DEALER INQUIRIES INVITED

- **Templates included:**

— Loan Amortization Schedule
— Net Present Value
— Future Value
— Personal Financial Statement
— Alternative Investment Analysis
— Check Register
— Personal Register (Roster)
— Depreciation Schedule

INVENTORY MASTER



"Software fit for a King!"

Royal Software

FEATURES:

- Over 1900 items per Double Density disk.
- Part number with Product Description.
- Quantity in stock/Number of orders outstanding.
- Three vendors with cost from each vendor.
- Last Purchase Order and from which vendor.
- Min/max re-order level.
- Last six months sales history.
- Retrieve any item within 5 seconds.
- Fast edit capability.
- Automatic Record updating from Purchase Orders and Product Invoices.

DEALER INQUIRIES INVITED

$89.95

Requirements
48K (min) Atari computer. At least one disk drive. 80 column printer.

REPORTS GENERATED:

- Purchase Orders
- Recommended Orders Report
- Inventory Control Report
- Product History Report
- Outstanding Orders by Vendor
- Monthly Sales and Inventory Level Totals
- Product Price List
- Period to Date Sales Report
- Daily Transactions Report

FREE CATALOG
with any order ... or send $1 (refundable with first purchase)

IF YOU DON'T HAVE OUR CATALOG... YOU'RE MISSING OUT!
NEW! 64 PAGE SUMMER ISSUE



DELUXE DUST COVERS ONLY **$8.95** EACH

Additional covers ordered at same time ONLY **$7.95** EACH



Custom fitted, attractive leather brown color:
• ATARI 400, 800, 600/800/1200XL, New XE & ST, 410, 810, 1050, 1025, 1027, CX85 • EPSON, GEMINI, PROWRITER printers • INDUS, RANA, PERCOM, TRAK disk drives.

Protect Your Equipment

Super Mailer PLUS

One of the most versatile data-base programs available.

- Maintain your book library
- Organize your record collection
- Index your recipes
- Categorize your stamp collection
- Unlimited applications!

48K Disk $49.95

Features:

- LIGHTNING FAST RETRIEVAL
- FAST SORTS ON ANY FIELD
- SUPPORTS UP TO 4 DRIVES
- SINGLE OR DOUBLE DENSITY
- STORE ABOUT 1100 RECORDS PER DISK SIDE IN DOUBLE DENSITY
- MUCH MORE!

Includes:

- ON-SCREEN PROMPTS
- HELP SCREENS
- STATE ABBREVIATION TABLE
- PRINT LABELS 1, 2, or 3-UP
- REDEFINABLE FIELDS
- MERGE-CREATE COMBINATION FILES
- DELETE DUPLICATE-AUTO or MANUAL

Super Merger

$19.95
48K, Disk

SuperMerger makes files from SuperMailer Plus compatible with any word processor (e.g., AtariWriter, Letter Wizard) that uses standard Atari DOS, and has a mail-merge function. This means you can design a form letter and send it to all the names on your SuperMailer Plus mailing list. SuperMerger also allows you to arrange the data from SuperMailer Plus in any sequence that is desired.

BOOKS

De Re Atari	$17.90
Technical Users Notes	$26.90
Kids & the Atari	$19.95
Best of Antic	$12.95
Best of Antic w/disk	$24.95
Machine Lang. f/Beg.	$14.95
2nd Bk. of Machine Lang.	$14.95
Your Atari Computer	$17.95
Master Memory Map (350 pg.)	$15.95
Master Memory Map (30 pg.)	$4.95
Adventures with the Atari	$16.95

PRINTERS

Epson LX-80	$259.95
Epson FX-80	$395.00
Star SG-10	$269.00
Axiom SLP	$199.95
SLP Tractor	$29.95
Axiom 550	$279.00
Prowriter 8510SP	$399.00
Prowriter 1550	$499.00
Atari 1025	$239.50
Atari 1027	$199.00
Atari 850 Interface	$139.50

— SUPER SPECIALS —

Gorf (C)	$ 5.95
Del. Invaders (C)	$ 5.95
Wizard of Wor (C)	$ 5.95
Krazy Shootout (C)	$ 5.95
Atari Speed Rd. (T)	$19.95
Invit. to Prog. #3 (T)	$19.95
Touch Typing (T)	$14.95
Baseball A.H. (T)	$15.50
Crossfire (T)	$ 9.95
Match Racer (D)	$ 9.95
Pathfinder (D)	$ 9.95
Miner 2049er (D)	$17.95
Tricky Tut. 1-12 (D)	ea. $ 7.95
Pharaoh's Curse (D)	$ 9.95
Ballblazer (D)	$25.95
Buck Rogers (D)	$25.90
Flight Simulator II (D)	$33.50
Pool 400 (C)	$14.95
Archon (D)	$19.90
Bk. of Atari SW '84	$ 9.95
Compuserve	$29.95
Percom S1 Drive	$159.95
130XE	$159.95
520 ST PACKAGE	$749.95

TRIVIA QUEST



- **Outsmart Your Friends**
- **Outwit The Dragon**
- **Join The Quest**

48K Disk
$39.95
- **Program Covers 4 Disk Sides**

A new concept in computer gaming. Intellectual challenge, strategy and arcade action. Each player assumes the role of a lord with a questing party of three characters. Complete the quest, earn the most gold by answering questions and battling the dragon. Win the favor of the king and thus, the game.

TRIVIA QUEST UTILITY EDITOR

$24.95
48K, Disk

The Trivia Quest Utility Editor is designed for convenient creation and editing of questions for Trivia Quest. Create new music and text questions, or change the questions that are already available. This program also includes another 1000 questions for use with the Trivia Quest program.

COMPUTER PALACE

OPEN M-F, 9-6 Sat. 10-4 (Pacific Time)
2160 W. 11th Avenue Eugene, Oregon 97402

USE YOUR CREDIT CARD & CALL
Toll Free 1-800-452-8013
★ ORDERS ONLY, PLEASE ★
There's never a penalty for using your credit card!
For Information, Call (503) 683-5361

SHIPPING INFO: Minimum $2.90 Ground, $4.75 Air. Actual Cost depends on weight. Call (503) 683-5361 for information.
WARRANTY INFO: Everything that we sell is warrantied by the manufacturer. If any item purchased from us fails to perform properly when you receive it, call us at (503) 683-5361 so that we can assist you. No returned merchandise accepted without authorization. Defective software will be replaced with another copy of the same program, otherwise, no software is returnable.

ONE-PASS DISK COPY 130

Antic's first 130XE program

by ERNIE NEGUS

Copy your disks in one quick pass using the full 128K memory of the 130XE. Features single read/multiple write—and write to multiple drives. Perfect for disk librarians. This fast, convenient copying program requires the new Atari 130XE computer, a disk drive, and a Disk Operating System that's compatible with DOS 2. It is written in MAC/65 assembly language. But you don't need to know machine-language, we have included a BASIC loader which will create the program for you.

Just bought a 130XE? Wondering what to do with all the extra memory? Me too. I wanted to get started with my new computer by writing a simple program that would use the larger memory, but I didn't know what to do. Then it occurred to me there was enough memory in my 130XE to hold 768 sectors of a single density disk. Single pass disk duplication guaranteed! It was like a dream come true.

All I needed was to write it. I started with a very simple three sector machine language program that simply did a one-pass duplication, switching the banks when needed. But after showing my program to friends and receiving lots of interesting suggestions, more and more features were added. The result is this powerful, multi-purpose sector copying program.

THE LISTINGS

Listing 1, COPY130.BAS, is a BASIC program that will create the binary load version of COPY 130. Type it in and check it particularly carefully with TYPO II. All that data is machine language and must be exactly right! Be sure and SAVE a backup copy before RUNning it.

When you RUN Listing 1, it will read the data and then ask for a device and filename. Place a formatted disk containing DOS 2 or DOS 2.5 in your drive and type D:AUTORUN.SYS [RETURN]. The file will be written to disk, after which you may simply boot the disk without BASIC to automatically RUN the program. Note that you should NOT have the RAMDISK activated if using DOS 2.5 when running this program. If you're using DOS XL from Optimized Systems Software, you can also enter another filename with a .COM extender and load the program from DOS.

Listing 2, COPY130.M65, is the assembly language source code and is included primarily for instruction. However, if you wish you can type it in instead of the BASIC loader. Some of you who are studying the use of the 130XE extended RAM will want to adapt portions of the program for your own use.

Antic Disk subscribers: Use DOS to transfer the file COPY130.EXE to another disk. Then rename it AUTORUN.SYS.

USING COPY 130

For most uses, the instructions that follow won't even be needed, just follow the onscreen prompts. You won't need to press [RETURN] at any prompt unless told to do so. Pressing [ESC] at any prompt will re-run the program. Those of you who are experienced at using sector copiers might want to skip to "Advanced

Usage" for the special features available at some prompts.

Before actually doing the copying, several prompts allow you to use any drive as a destination drive, format a destination drive, verify writes and handle possible errors. There is no provision for indicating a source drive. I intentionally left this out because everyone I know uses drive 1 as the source drive.

The "Destination #" prompt asks which drive (1—8) to use for the destination disk. If [1] is entered, you'll receive an "Insert Destination Disk" prompt after the source disk is read into memory. Any other number will skip this prompt so you must have your destination disk properly inserted at the "Insert Source Disk" prompt.

The "Format ?" prompt asks if wish to format your destination disk(s). Unless the disk you will be copying to is already formatted, answer this prompt by pressing [Y]. Any other key yields a No. The "Verify ?" prompt is asking if the program should verify each sector write by re-reading and comparing each sector. A [Y] here will make the copy take longer, but will insure that the data written is valid.

"Continue on Error" is for duplicating a disk with damaged sectors. Unless you know the disk you are reading is damaged, answer [N] to this prompt, otherwise the program will not stop if an error occurs.

DATA WINDOW

After inserting the source disk and pressing [RETURN], the program will begin reading your disk into RAM, and a data window will appear on the screen.

The left-most number in this window is the current drive number. The right-most window (which should contain nothing) is the error window. The center window is the number of the sector currently being processed. If the program stops before the sector number reaches 720, an error has occurred and a number will appear in the error window. See *Figure 1* for error numbers. If no errors occur, you are prompted to insert a destination disk in the selected drive. Do this and then press [RETURN].

FIGURE 1

ERROR CODES

Number	Meaning
0	Non-I/O error (Serious Error—Program Damage).
1	Break key was pressed, causing loss of data.
2	Timeout- Device doesn't respond.
3	NAK- No acknowledge from device.
4	Serial Bus Data error.
5	Write-Protect, open drive or bad sector.
6	Read after write verify error.
7	Unrecoverable system I/O error.
8	Damaged sector links.
9	Bad Sectors found at format time.

When you use this option at the "Destination #" prompt, the destination defaults to drive 1. Set up the variables as explained before, then load the source disk. When you get to "Insert Destination Disk", load each drive you have—including drive 1—with blank disks. The program will then write the source information to each drive automatically without prompts. Answering [Y] to "Another Copy ?" will go back to the "Insert Destination Disk" prompt where you can reload and write to all your drives again.

When the copy is finished you are prompted for another copy. If you want to make another copy of the source disk already in RAM, press [Y], otherwise press any other key.

ADVANCED USAGE

When you press [Y] at the "Another Copy ?" prompt, the program uses the same drive, format, verify and error handling variables that were previously set. To change these without reloading the source disk, press [ESC] at this prompt, reset your variables, but when you come to "Insert Source Disk— Type Return", press [N]. This skips the reloading of the source disk, and can also be used if an error has occurred when writing or formatting and you wish to try again.

Three other keys are recognized at the "Destination #" prompt. These are [D], [T] and [Q] which are used to enter the multi-destination mode of operation. Multi-destination mode allows users who own more than one drive to efficiently mass-produce disk programs by writing to multiple drives. [D] is for Double—two drives—[T] is for Triple and [Q] is for Quadruple.

Ernie Negus of Portland, Oregon is a regular ***Antic*** *contributor, specializing in machine language programming.*

Listing on page 55.

". . .enough memory in my 130XE to hold 768 sectors of a single density disk."

16-BIT SOUNDPOWER

How the pros enhance Atari music

by JERRY WHITE

Beef up your Atari music from BASIC. Learn how to program 16-bit dual-voice sound. You'll get a richly melodic seven octaves, instead of the thinner-sounding four octaves you'd hear otherwise. This short BASIC program works on any Atari computer of all memory sizes, with disk or cassette.

If you have experimented with SOUND commands in Atari BASIC, you probably noticed that some of the higher notes seem a bit flat. You may have also found that your lowest note is the B generated by SOUND 0,255,10,8.

Using SOUND commands with a distortion value of 10 for clear sound, you have a range of just over four octaves. If you'd like to fine-tune your music and extend that range to seven octaves, this tutorial will tell you how.

The SOUND 16 program will demonstrate what is called 16-bit sound. It is based on using two combined voices to create one sound. Only two sounds can be produced at once, but the frequency of each pitch will be more accurate and much deeper bass notes can be generated.

SOUND16

Type in Listing 1, SOUND16.BAS, check it with TYPO II and SAVE a copy before you RUN it.

This listing uses an assembly language subroutine to turn 16-bit sounds on and off. This assembler routine appears in the BASIC program as the DATA statements starting at line 20010. The FOR/NEXT loop at line 130 reads this data and stores our subroutine in the string S16$.

This demonstration program shows how the desired frequency and volume for one or two voices can be passed to the subroutine by using BASIC's USR function. Note that the SOUND and POKE commands found in line 150 must be executed before your first USR call.

The BASIC program reads frequency data into an array called FREQ. This array stores 12 frequencies for each of seven octaves. Octave one contains the highest note frequencies, while octave seven contains the lowest bass notes.

Each octave begins with C as its lowest note (pitch 12), and ends with B as its highest note (pitch 1). Middle C (SOUND 0,121,10,8) is frequency 3414 or FREQ(4,12) in our array. The next highest note, C#, is FREQ(4,11). The next lowest note, B, is FREQ(5,1).

BEAUTIFUL MUSIC

The program uses a countdown timer to clock delays. When you POKE a number from 1 to 255 into location 540, it will be decremented every 1/60th of a second. 1/60th of a second is called a "jiffy." Thus, if you set the variable WAIT=60, then go to the subroutine beginning at line 480, you will return in one second.

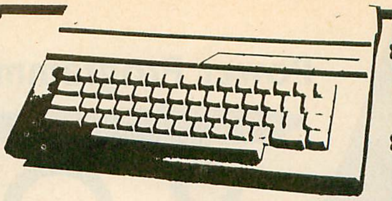
Octave, pitch and note will be displayed on the screen as the program cycles through all frequencies. Next, the double 16-bit sound option will be used to demonstrate the use of consecutive octaves. Finally, a short tune is played just before the program ends.

Listing on page 65

Antic Contributing Editor Jerry White is a leading light of Atari programming and founded PAPIE, the Professional Atari Programmers Information Exchange.

ATARI

POWER WITHOUT THE PRICE AT. . .COMPUTER CREATIONS



ATARI HARDWARE PACKAGES

Pkg. #1
Atari 520 ST Computer
Atari SM 124 Monochrome Monitor
Atari SF 354 (3½") Floppy Disk Drive

Pkg. #2
Atari 520 ST Computer
Atari SC 1224 (12" RGB) Color Monitor
Atari SF 354 (3½") Floppy Disk Drive

Pkg. #3
Atari 800 X/L Computer
Atari 1050 Disk Drive

Call For Our Low Prices

DISK DRIVES
Atari 1050
Indus GT (Free Software) .... CALL
Atari SF 354 (3½" floppy disk 500K) .... FOR
Atari SH 317 (3½" hard disk 10 MB) .... NEW PRICES
Happy Enhancement for Atari 810 and 1050 Drives

PRINTERS
Atari XTM 201
Atari XTC 201 .... CALL
Atari XDM 121 .... FOR
Atari XMM 801 .... NEW
Atari ST 504 .... PRICES

EPSON PRINTERS
LX-80 (80 column)
LX-80 Tractor Feed
Epson FX-80+ (80 col.)

STAR MICRONICS PRINTERS
SG-10 (80 column)
SG-15 (136 column) .... CALL
SD-10 (80 column) .... FOR
SR-10 (80 column) .... PRICES
Powertype Daisywheel
Powertype Tractor Feed

PANASONIC PRINTERS
KX - 1090 .... CALL
KX - 1091 .... FOR
KX-1092 .... PRICES
Okimate 10 plus plug n' print

PRINTER INTERFACE CABLES
MPP-Microprint 39
MPP-1150 Parallel Printer Interface 59
U-Print A 69
A-16 Interface/Buffer 99
APE Face XLP 59
APE Face 12XLP 59
Microbits Microstuffer 109
MPP-1151 62

PRINTER RIBBONS
Gemini Printers (Black) 3
Gemini Printers (Blue/Red/Purple/Brn./Grn.) 4
Epson Printers (80) Series) 6

FLOPPY DISKS AS LOW AS 79¢ EACH
DISKETTES (2 box minimum) 10 per box

	Generic (SKC)		Wabash		Maxell		Generic (Bulk)		BASF (Ea.) 3½"	
# Boxes	SS/DD	DS/DD	SS/DD	DS/DD	SS/SD	DS/DD	SS/DD	DS/DD	SS/DD	DS/DD
2	9.99	11.99	13.99	19.99	18.99	27.00	7.99	9.99	3.49	4.99
3 - 6	8.99	10.99	12.99	19.49	17.99	26.00	6.99	8.99	3.19	4.49
7 - 10	7.99	9.99	11.99	18.99	16.99	25.00	6.49	8.49	2.99	3.99

Panasonic Printers, Black 8
Panasonic Printers, Color 11

MONITORS
Atari XC 141 (14" Composite Color)
Atari XM 148 (12" Monochrome, 80 column, low resolution) .... CALL FOR PRICES
Atari SM 124 (12" Monochrome, 80 column, high resolution)
Atari SC 1224 (12" RGB Color)
Sanyo 12" Green Screen 79
Sanyo 12" Amber Screen 79
Monitor Cable 5
Teknika 13" Color MJ 10 189

MODEMS
Atari 1030 Direct 300 BAUD 69
Atari XM 301 Direct Connect 300 BAUD .... CALL
MPP-1000E Modem 79
Signalman XII Modem 199
R-Verter 39
Compuserve Starter Kit 21
850 Interface 114
Anadex (Hayes Compatible) 239

UPGRADES/ACCESSORIES
Flip n' File 10 4
Flip n' File 15 7
Original Flip n' File 50 14
Flip n' File Cart Case 12
Library Cases(10 colors) 4
Disk Bank/5 (Holds 50) 12
Disk Bank (Holds 10) 5
Power Strip (6 outlet) 16
Lineguard Spike Suppressor 13
Disk Drive Cleaning Kit 6
MicroMate Paper (20#, 540 sheets.) 10
Printer Stand (wire) 16
Dust Covers .... Call for availability
Disk Coupler (notch) 3

UPGRADES:
Ram Rod X/L W/Omnimon 99
Omniview for 800 XL 49
U.S. Doubler 59
MPP 64 K 69
R-Time 8 Cartridge 59

COMPLETE LINE OF ATARI SOFTWARE
CALL IF YOUR PROGRAM IS NOT LISTED!!

SOFTWARE

NEW ATARI PROGRAMS
Codewriter 39
Filewriter 21
Reportwriter 21
Menuwriter 21
Home Integrator 20
Small Business Inventory 12
Salesman's Expenses 12
Accounts Rec./Pay. 12
Retail Invoice 12
Final Legacy 15
Adventure Writer 18
Star Voyager 12

EPYX
Dragonriders of Pern (D) 21
Summer Games 28
Pitstop II (D) 28
Ballblazer (D) 28
Rescue on Fractalus (D) 28
Temple of Apshai (D) 21

ACCESS
Beach Head (D) 24

ACTIVISION
Decathalon (R) 18
Pitfall II (R) 18
Space Shuttle (R) 18
Ghost Busters (D) 22

BATTERIES INCLUDED
Paperclip (D) 41
Homepak 35
B/Graph 48

BRODERBUND
Printshop (D) 31
Spelunker (D) 21
Stealth (D) 21
Whistler's Brother (D) 21

DATASOFT
Bruce Lee 28
Dallas Quest 24
Conan 28
Letter Wizard 52

ELECTRONIC ARTS
Archon (D) 18
Pinball Construction (D) 18
M.U.L.E. (D) 18
Murder/Zinderneuf (D) 18
One on One (D) 25
Archon II (D) 25
Music Construction (D) 18
Realm/Impossibility (D) 18
Hard Hat Mack 14
Seven Cities of Gold 25
Cut and Paste 18

INFOCOM
Cut Throats (D) 24
Deadline (D) 31
Enchanter (D) 24
Hitchhiker's Guide to the Galaxy (D) 24
Infidel (D) 28
Planetfall (D) 24
Sea Stalker (D) 24
Sorcerer (D) 28
Starcross (D) 31
Suspect (D) 28
Suspended (D) 31
Witness (D) 24
Zork I (D) 24
Zork II or III (D) 28
Invisiclues Hint Books 7

MICROPROSE
F-15 Strike Eagle (D) 24
Mig Alley Ace (D) 24
Solo Flight (D) 24
Kennedy Approach 24

OSS
Action (R) 62
Action Tool Kit (D) 25
Basic XL (R) 50
DOS XL (D) 25
MAC/65 (D) 50
Mac 65 (R) 62
MAC/65 Tool Kit (D) 25
Writer's Tool Kit 62

ORIGIN
Ultima III (D) 42

SCARBOROUGH
Mastertype 28
Net Worth 55

SIERRA ON LINE
Ultima I 24
Ultima II 41

SYNAPSE
Quasimodo 17
Alley Cat 14
Syn-File + ... 35
Syn-Calc 35
Syn-Trend 28
Syn-Comm 28
Syn-Stock 28
Mindwheel (needs 2 drives) ... Call

TRONIX
S.A.M. 42
Chatterbee 28

To order call TOLL FREE
1-800-824-7506
ORDER LINE ONLY

No extra charge for MasterCard or Visa

COMPUTER CREATIONS, Inc.
P.O. BOX 493 - DAYTON, OHIO 45459
For information, order inquiries, or for Ohio orders (513) 435-6868

Order Lines Open 9:00 a.m. to 9:00 p.m. Mon.-Fri.; 10 a.m. to 4:00 p.m. Sat. (Eastern Standard Time). Minimum $15 per order. C.O.D. (add $3.00). Please specify computer system. Call toll free number to verify prices and availability of product. Prices and availability are subject to change without notice. We ship C.O.D. to Continental U.S. addresses only! Hardware requires additional freight charges (min. $4.00). Software and accessories add $3.00 shipping and handling in Continental U.S. Actual freight will be charged outside U.S. to include Canada, Alaska, Hawaii, Puerto Rico and APO. Ohio residents add 6% sales tax. Canadian orders add 5% shipping, (min. $5.00). All other foreign orders, please add 15% shipping, (min. $10). For immediate delivery send cashier's check, money order or direct bank transfers. Personal and company checks allow 3 weeks to clear. School purchase orders welcome. Due to our low prices, all sales are final. NO CREDITS. All defective returns must have a return authorization number. Please call (513) 435-6868 to obtain an RA# or your return will not be accepted for replacement or repair.

8 QUEENS ACTION!

92 chess solutions in 40 seconds

Program by Dave Oblad

Lightning-fast ACTION! solution to "The Eight Queens Problem" from the April, 1985 ***Antic****. Requires the ACTION! programming language cassette from Optimized Systems Software. Works on all 8-bit Atari computers of any memory size, with disk or cassette. Disk subscribers: You can use this program without ACTION! Select the "L" option from DOS 2 for the file, QUEEN.EXE.*

In line with **Antic's** long-held belief that our published programs are part of a two-way communications process with readers of the magazine, many **Antic** programming take-aparts conclude with suggestions for possible enhancements that an ambitious programmer might make in the listings.

But Dave Oblad took it as a personal challenge when he saw Angelo Giambra's "The Eight Queens Problem" in the April, 1985 **Antic** and read our final comment: "For a real challenge, you might want to try modifying the program so that only the 12 *unique* solutions are found."

(The original April article showed that there are 92 possible ways to arrange eight queens on a chessboard so that none of them threatens another. As the most powerful chess piece, a queen can attack for any distance along any straight line.)

In Dave's letter to **Antic**, he wrote, "I spent the next two days cranking away at my Atari in the ACTION! language, which is much faster than BASIC. My algorithm solves and displays all 92 general solutions in approximately 40 seconds—and finds the 12 unique solutions in 30 seconds."

Listing on page 66

PLOTTERS FOR YOUR ATARI

Graphics hardcopies in color

by ERIC CLAUSEN

It's a vicious circle. . . . In fact sometimes it seems like an evil plot! (Sorry, couldn't resist.)

There hasn't been very much plotter hardware or software available for the Atari. And the few choices available are not widely publicized. So Atari owners naturally don't use plotters a great deal.

What's especially sad about this is most Atari owners don't even know that now they *can* use plotters to create colorful graphics in very high resolution.

Only in schoolrooms have Atari computers been widely used with plotters—typically with the narrow-width but capable **Atari 1020 Color Plotter**. (See adjoining story about 1020 availability.) This setup then runs **Atari Logo** software to produce beautiful hardcopy screen dumps of turtle graphics. Giving students colorful printouts of their classroom computer exercises can obviously enhance the learning experience.

WHAT'S A PLOTTER?

A plotter draws smooth, continuous lines with ballpoint pens or felt-tip pens that glide silently over the paper. Many plotters can be programmed to change pen colors automatically. Plotters generally can also be used to produce overhead transparencies for educational or business use.

Until recently, most Atari plotter software was in the public domain—useful enough programs but tending to lack the sophisticated polish of professional packages. It also was not easy to get hold of this software unless you were in contact with a users group or bulletin board.

But now several very sophisticated graphics packages have become available which support a number of plotters with the Atari.

PLOT, GRAPH & 1020

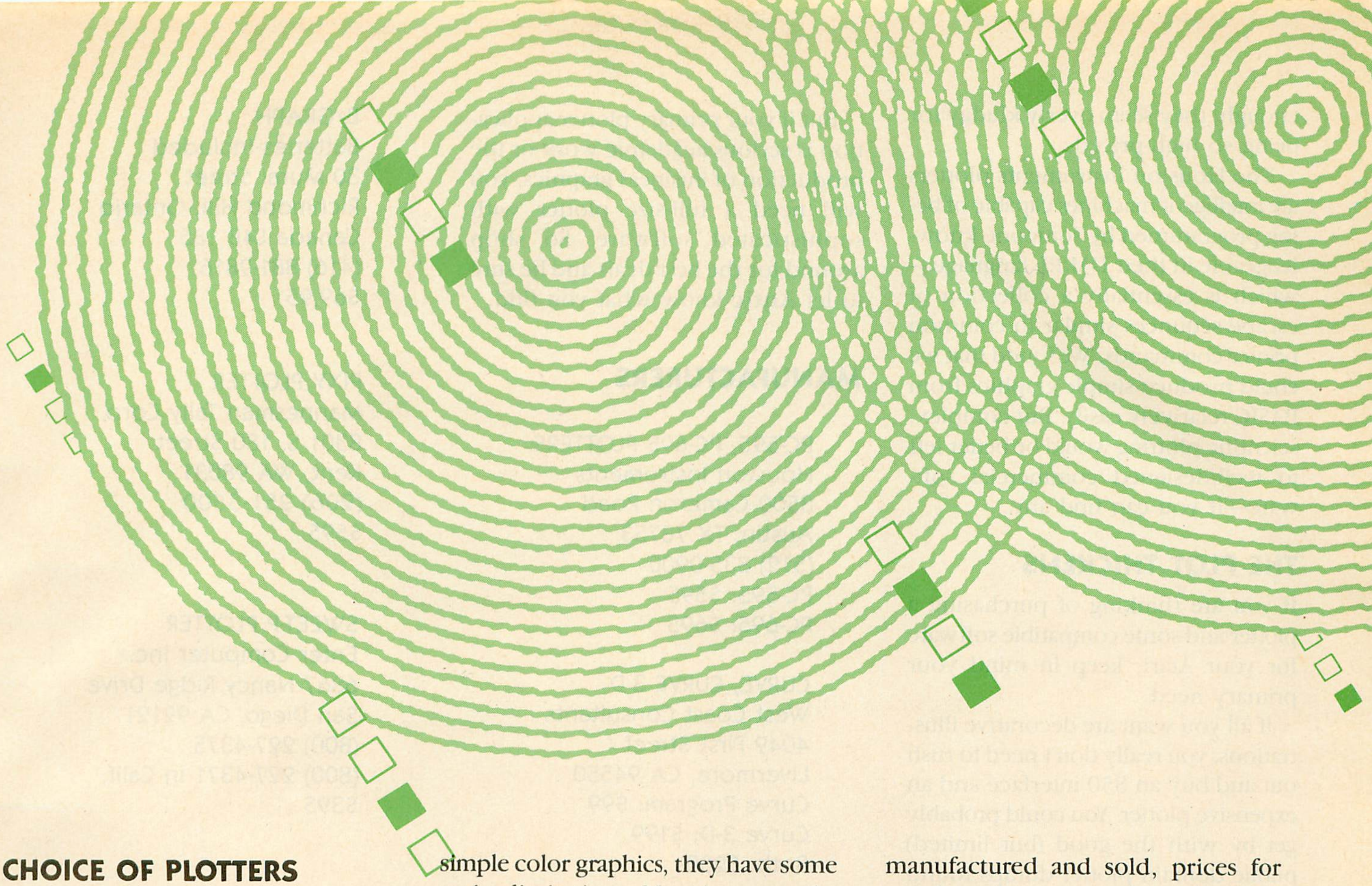
Dollar for dollar, the best Atari plotting value we found is **Screen Plot** and **B/Graph** software with the 1020 plotter! It's very satisfying to see an actual hardcopy of a microscreen that you've labored over for hours. And it's a whole new ballgame when you can print standard business graphics in flashy multi-colors.

Screen Plot is a plotter driver by Robert Wilson and Batteries Included's Michael Reichmann. It costs $12.95 from the Antic Arcade Catalog (AP135). This easy-handling but versatile graphics dump program accepts files from a wide range of graphics software—including Micro Illustrator, AtariArtist, Atari Light Pen, Paint, MicroPainter, or your own GRAPHICS 8 files.

Last but not least, Screen Plot works with B/Graph ($69.95) the full-featured statistical graphing package from Batteries Included. This isn't surprising, because the authors of B/Graph also wrote Screen Plot. The two programs work together in an ideal partnership. B/Graph generates bar graphs, pie charts and line graphs on your video display. And then Screen Plot dumps the files to a color plotter.

The B/Graph disk has a BASIC file conversion utility called PICLOAD.BAS. After you image a B/Graph file to disk, LOAD PICLOAD.BAS from DOS and follow the menu prompts. Screen Plot will then be ready to dump your creation to the color plotter.

NOTE: If you bought B/Graph version 1.0 from Inhome Software instead of the more recent Batteries Included version 1.1, your disk does not contain PICLOAD.BAS. However, PICLOAD.BAS is in public domain and can be downloaded from DL4 in SIG*Atari on CompuServe.



CHOICE OF PLOTTERS

The plotters supported by Screen Plot software are the Atari 1020, **Radio Shack GCP-115, Mannesman Tally Pixy** and the **Sweet- P**.

The Atari 1020 and Radio Shack's CCP-115 were both produced by the same OEM (Original Equipment Manufacturer). They are virtually the same piece of equipment. For an Atari owner, the main difference is that the 1020 daisy chains right into one of your disk drive ports. But the Radio Shack requires connection to the parallel port of the Atari 850 Interface Module.

Although both of these plotters have been discontinued, thousands of units are still available at excellent prices.

It's easy and inexpensive to get an Atari 1020 by mail. See the adjacent list of suppliers, for prices as low as $39.

You can usually find a few 115's at the nearest Radio Shack store, at prices ranging from $99 to $119. Radio Shack is also a handy source of refill pens, usually at a little over $2 for a package of four colors.

But although the 1020 and the Radio Shack plotters do a nice job on simple color graphics, they have some major limitations. Most importantly, they only work with narrow 4½" paper and they only give you four colors!

Also, the ballpoint pens available for the 1020 and the 115 are messy to work with, have a short life expectancy and tend to produce uneven ink flow during the often lengthy time of a plotting session.

UPPING THE ANTE

On the other hand, the Mannesman Tally Pixy is a full-width, 8-color plotter. This would be a superb choice of hardware if you want to do serious academic and business applications on your Atari with Screen Plot and B/Graph, However, it is list-priced at $595 and requires the 850 interface.

The Sweet-P SP100 is a lower cost alternative to the Mannesman Tally. It's a full-size, one pen plotter reasonably priced at $395. Again, this also requires the 850 interface. The SP100 also has a very impressive big brother, the SP600. This is a fast, six color, full-size plotter—at the full size price of $1,095.

The unfortunate fact is that because comparatively few plotters are manufactured and sold, prices for good equipment tends to be high compared to dot-matrix and daisy-wheel printers.

Antic had the opportunity to check out two programs that connect an Atari to the professional-quality **Houston Instruments PC-595** ($595) and **PC-695** ($695) These impressive full-size plotters naturally need the 850 interface. And they also needed better software for their Atari hookup. . .

The programs, **Curve** ($99) and **Curve 3-D** ($199) are written entirely in slow, slow BASIC. Many of the mathematical calculations are performed on screen, perhaps to convince the user that the computer is actually doing something. Someone should have told the programmers they could have gotten a lot more speed by merely blanking the screen with a simple POKE.

Also the data input prompts are filled with irritating sound effects and silly graphics. Although these programs are graphically ambitious and do eventually produce some impressive printouts if you wait long

continued on next page

enough, they seem overly kludgy for being so high-priced.

The Houston Instruments plotters are intelligent machines capable of being programmed by someone knowledgeable in their DM/PL language—which is essentially a collection of escape sequnces similar to common printer commands. We found that we could produce simple graphics from BASIC relatively easily. But complex, scientific plotting would be better left to well-designed commercial software—if you can find any.

THE PLOT THICKENS

If you are thinking of purchasing a plotter and some compatible software for your Atari, keep in mind your primary need.

If all you want are decorative illustrations, you really don't need to rush out and buy an 850 interface and an expensive plotter. You could probably get by with the good (but limited) public domain plotter dumps found on CompuServe's SIG*Atari.

But if your primary plotter application is serious academic work or the production of business graphics, you will need a full-size plotter with sophisticated software. So shop around for the best deals and be sure to let **Antic** know what you find.

MANUFACTURERS

PC-595, PC-695 PLOTTERS
Houston Instruments
8500 Cameron Road
Austin, TX 78753
(512) 835-0900
PC-595: $595
PC-695: $695

CURVE, CURVE 3-D
West Coast Consultants
4049 First Street
Livermore, CA 94550
Curve Program: $99
Curve 3-D: $199
Both: $250

B/GRAPH
Batteries Included
30 Mural Street
Richmond Hill, Ontario
Canada L4B 1B5
(416) 881-9816
$69.95

PIXY PLOTTER
Mannesman Tally Corp.
8301 S. 180 Street
Kent, WA 98031
(206) 251-5500
$595

SWEET-P PLOTTER
Enter Computer Inc.
6867 Nancy Ridge Drive
San Diego, CA 92121
(800) 227-4375
(800) 227-4371 in Calif.
$395

ATARI 1020 COLOR PLOTTER: WHERE TO FIND IT

The Atari 1020 Color Plotter is another pre-Tramiel product that's usually not easy to find at your local computer store—even though Atari has told **Antic** it still has plenty of units warehoused.

In order to help readers get around this unavailability, **Antic** is printing here a list of dealers offering the 1020 plotter by mail at varying prices. Most of the companies named here were also on last month's list of Atari 850 interface suppliers.

American TV
15338 Inverness Street
San Leandro, CA 94579
(415) 352-3787
$54.50

B&C Computervisions
3283 Kifer Road
Santa Clara, CA 95051
(408) 245-2680
$60

CompuCat
24500 Glenwood Highway
Los Gatos, CA 95030
(408) 353-1836
$59.73, limited supply

CompuClub
P.O. Box 652
Natick, MA 01760
(800) 631-3111
$65

Computer Creations
P.O. Box 292467
Dayton, OH 45429
(800) 824-7506
$39

Computer Palace
2160 W. 11th
Eugene, OR 97402
(800) 452-8013
$99

Electronic One
P.O. Box 13428
Columbus, OH 43213
(614) 864-9994
$39.77

Gizzmoz
P.O. Box 1402
Concord, MA 01742
(800) 225-5800
$49.95

San Jose Computer
1844 Almaden Road
Unit E
San Jose, CA 95125
(408) 723-2025
$39

Southern Software
1579 Ruffner Road
Birmingham AL 35210
(205) 956-0986
$79.95



Computer heroes or criminal vandals?

HACKERS FOREVER!

by NAT FRIEDLAND, **Antic** Editor

As you may have noticed, the word "hacker" does not appear very often in **Antic** Magazine. That's because there are at least three conflicting definitions of hackers and it can get very confusing.

Personally we're most comfortable with the earliest good-guy definition—hackers simply as skilled, intensely committed programmers.

But not long afterward, elements of the "nerd" lifestyle got added onto the definition. Hackers were often thought of as being socially inept and completely losing track of time during their marathon sessions at the computer.

And of course in recent years the public has come to see hackers primarily as maliciously mischievous teenagers who break into high-security mainframe computer systems and vandalize them. In other words, the "WarGames" movie definition.

In this essay we will look at three major new books that use all these conflicting hacker definitions and more. . . And ultimately we'll seek to understand what hackers have *really* meant in the revolutionary development of personal computing.

PART I: HERO HACKERS

HACKERS
by Steven Levy
Anchor Press/Doubleday
501 Franklin Avenue
Garden City, NY 11530
458 pages, hardcover
$17.95

Steven Levy covers technology for "Rolling Stone" and he has taken the high road in his evaluation of hacking. The subtitle of his book ***Hackers*** clearly shows his viewpoint—"Heroes of the Computer Revolution."

The approach is in the tradition of Tom Wolfe's *The Right Stuff* and Tracy Kidder's *Soul of a New Machine.* There's a kind of breathlessly amazed quality to the writing, as the style pushes to reveal significant meanings out of gritty real-life details.

Hackers is divided into three parts. The first (and best) part covers the dawn of hacking at Massachusets Institute of Technology starting in the late '50s.

Getting started as a hacker in those days was not as simple as buying (or getting your parents to buy) a personal computer.

The only computers in existence were huge, hulking, air-conditioned mainframes that were ridiculously crude and puny by today's standards. The conversion from punch-card batch programming to more interactive video terminals was just getting underway.

In 1961, Digital Equipment Company's second manufactured PDP-1 unit showed up on the ninth floor of MIT Building 26. This was the first true minicomputer and its price was an astoundingly low $120,000.

But even before that, the hacker subculture was well established at Cambridge's Tech Square. Some of the nation's brightest students of science and engineering had converged on MIT and found themselves hypnotically attracted to computer programming as an emerging art form.

The only way they could get at the PDP-1 and its ancestors was to sign up for late-night time slots after the "official" graduate students had gone home. The only way to learn how to program was by looking over the shoulder of a more experienced hacker and asking questions.

continued on next page

The programs that needed to be written at this point were the basics of machine operation—screen editors, assemblers. And then came prototypes of chess programs, music programs, and the breakthrough game Spacewar.

The fascination of creating these new software forms was so overwhelmingly satisfying to the hackers that they literally put the rest of their lives on hold for years. Many of the MIT hackers could never be bothered completing their degree requirements.

However, the most talented of the hackers were quickly named to the staff at the MIT computer center and went on to have distinguished careers as professional programmers—particularly in the development of Artificial Intelligence.

Author Levy sees the MIT hackers as computer Johnny Appleseeds, moving on to other universities like Stanford and spreading the purest form of Hacker Ethic. The key elements of this ethic include:

- Access to computers and information should be unlimited. How else can you fix things that need improvement?
- Computers can create beautiful art and improve life.

WOZ WITHOUT JOBS

The middle third of *Hackers* deals with Northern California's "Hardware Hackers" of the '70s. The main theme is how Steve Wozniak created the first Apple microcomputer because he was so inspired by the free-swinging meetings of Lee Felsenstein's Homebrew Computer Club.

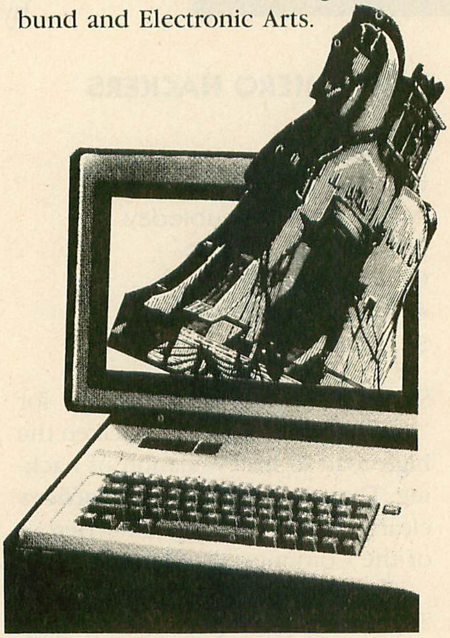
Frankly, this material is handled with more depth in *Fire In The Valley*, ($9.95, Osborne, McGraw-Hill) which was **Antic's** favorite general computer book of 1984 and devoted a full 288 pages to "The Making of the Personal Computer" in Silicon Valley.

For example, *Hackers* hardly mentions Wozniak's partner Steve Jobs, presumably because the author sees Jobs as some kind of high-tech hustler rather than a hacker. However, even this section of the book contains many new details and is well worth reading.

MONEY IN EDEN

The final section shows how the sky-rocketing popularity of the Apple II created the first large-scale market for computer software. And big money destroyed the purest form of the Hacker Ethic, even though there were still great programming feats to come.

This story is told mostly through Ken and Roberta Williams of Sierra On-Line, creators of the first graphics adventure games—**Mystery House** and **Wizard And The Princess**. Although Ken Williams was as much of an entrepreneur as a programming whiz, he and his shy, game-writing wife Roberta were not exactly into the traditional business ethic. In 1981, Sierra On-Line even organized a white-water rafting trip for the bosses of all the competing entertainment software houses, including Broderbund and Electronic Arts.



ATARI WHIZ KID

Antic readers will be particularly interested in the tale of Sierra On-Line programmer John Harris, a nice-looking but awkwardly shy San Diego teenager who earned over $100,000 a year as the first star independent programmer of games for Atari computers.

The Harris masterpieces were the Atari version of **Frogger** and the superior **Pac-Man** clone, **Jawbreaker**. Harris turned down a huge Atari offer for Jawbreaker because he wanted to punish the company for withholding information about the memory map.

Despite his negative feelings about the old Atari management, Harris loved the outstanding graphics and sound capabilities of his model 800. He had such contempt for the Apple that "At the very mention of the machine, Harris would recoil and make the sign of the cross, as if warding off a vampire."

In fact, Harris eventually quit Sierra On-Line and joined Atari-oriented Synapse Software largely because he believed Sierra was downgrading the Atari in favor of the Apple.

PART II: "DARK SIDE" HACKING

OUT OF THE INNER CIRCLE
by "The Cracker" Bill Landreth
Microsoft Press
10700 Northup Way, Box 92700
Bellevue, WA 98009
230 pages, trade paperbound
$9.95

The intention of this book is also given away by its subtitle, "A Hacker's Guide To Computer Security." ***Out Of The Inner Circle*** represents the grand old literary tradition of a "reformed" lawbreaker warning the public against his former gang.

Authors of this kind of book are rarely able to avoid giving the impression that they had much more fun before they went straight.

Inner Circle is an extremely well designed large-format paperback from the Microsoft software company. Interestingly enough, according to *Fire In The Valley*, Microsoft founder Bill Gates started out as a teenage large-system hacker before hitting it big in commercial programming.

Bill Landreth, "The Teenage Computer Wizard Who Was Apprehended by the FBI," is at his best when describing what motivates the kind of online hackers who devote their talent and time to gaining unauthorized entry onto large telecommunications systems.

According to Landreth, whose hacker bulletin board "handle" was The Cracker, most online hackers are not destructive. They are usually "Tourists" seeking the challenge of solving puzzles, or "Students" like himself who prize computer knowledge for its own sake.

Landreth writes that a high-level online hacker like one of his fellow members of the Inner Circle network "would never intentionally damage a system. He spent as much as 40 hours just to get access, and he wants to remain undiscovered so he can keep using your system. Besides, someday he may want to apply for a job as a system operator with your company. . ."

He complains that the tiny minority of crashers "give all hackers a bad name. They close down accounts that other hackers spend much time and effort to get."

This book gives candid critiques of the attractiveness and vulnerabilities of popular operating systems that run on large computers such as DEC, VAX and IBM models. Landreth also reviews much of the online security equipment on the market.

Particularly valuable are the author's descriptions of online hacking techniques. The Hack-Hack method would consist of autodialing possible passwords in every combination of letters. This is guaranteed to catch the notice of an alert sysop and is considered overly crude.

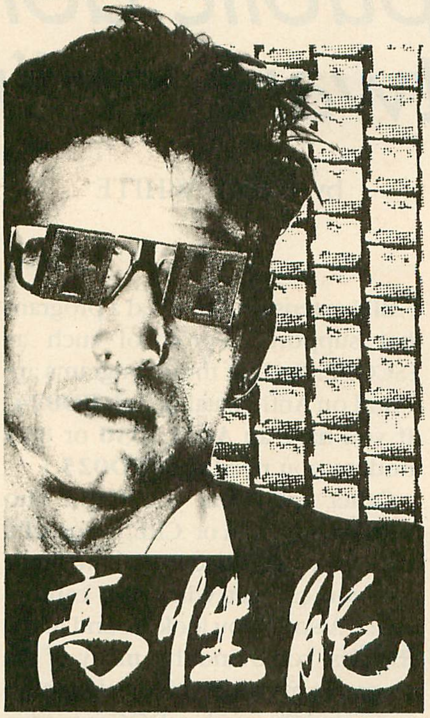
A more experienced online hacker would prefer subtler approaches such as programming a Decoy screen display and tricking legitimate users into leaving their passwords. This technique is why CompuServe always warns subscribers never to type in their passwords while participating in a CB conference.

Some of the other hacking techniques discussed are the Trapdoor (made famous in "WarGames"); the Trojan Horse, an inviting file that users would have to log onto with their passwords; Rapid-Fire buffer command shifts; and the dangerous Logic Bombs.

Landreth says he doesn't do online hacking any more, in the aftermath of plea bargaining with the FBI for three years' probation and the return of his computer equipment.

There were originally three counts of Wire Fraud against "The Cracker." Landreth insists that helping create an unauthorized bulletin board for hackers on the GTE Telemail system did not actually damage it.

But unfortunately the number of hackers using GTE soon grew to nearly 200 and a few of them carelessly left traces online. GTE panicked and called in the FBI.



PART III: SCI-FI HACKING

NEUROMANCER
by William Gibson
Ace Science Fiction
200 Madison Avenue
New York, NY 10016
271 pages, paperbound
$2.95

Case was only 24 and he had been one of the best interface cowboys roaming the computer matrix!

Jacked into a customized cyberspace navigation deck, Case used biofeedback cues and keyboard commands to project his disembodied consciousness into the earth-spanning digital matrix. He carefully made his way through the deadly ICE security software and stole confidential data for corporate spies.

Then he double-crossed one of his employers and was punished by injection with a nerve poison that burned out his talent. It was a revenge worse than death. Now Case's mind couldn't fly through cyberspace any more and he was trapped in the meat of his body. . .

At this writing, ***Neuromancer*** was considered a top contender for the year's major science fiction awards, the Hugo and Nebula. Certainly the quality of William Gibson's imaginative writing is very high. This first novel reads like a wildly hyped-up version of "Blade Runner" and is set in a convincingly gritty and dangerous near-future where computer crime takes on a whole new meaning.

. . .Case gets his talent restored by Tokyo underworld microsurgeons in return for taking part in what turns out to be a battle between two of the world's most powerful Artificial Intelligences, Wintermute and Neuromancer.

Dodging the AI police, Case teams up with Molly, a female street samurai who has silver radar eyes and retractable fingernail razors. Prior to the big caper, Case and Molly must steal well-guarded microchips containing the taped personality of his dead former mentor, the Dixie Flatline, who had been "hard-wired into ROM."

To help them, they hire the Panther Moderns, a gang of teenage terrorist merceneries who get high on software tubes plugged into implanted sockets behind their ears. And so it goes. . .

The science fiction vision of *Neuromancer* projects vivid images of a rapidly changing technology that creates human effects both disturbing and intriguing. Even if only a few of author Gibson's predictions come true, there could still be many great adventures waiting for hackers.

Getting back to today's world, it's true that persons labelled as hackers sometimes get carried away and pick up a bad name from the public. But it also seems true that much of what we know as the Computer Revolution really could not have happened without the self-motivated explorations of hackers in the '60s, '70s and '80s.

SIG*ATARI'S GREATEST HITS!

Best public domain software online!

by JERRY WHITE

The **CompuServe** SIG*Atari Data Libraries contain over 1,000 public domain programs. There is no cost for downloading these programs beyond the normal CompuServe access time charges.

To help you choose which programs you want from this treasure trove, I asked SIG members for recommendations. Special thanks to sysop Richard Brudzynski and the many other SIG members who gave time and suggestions.

GETTING STARTED

Downloading can be confusing at first, but you'll find fast help by leaving your questions for SIG members in a message. The Antic Arcade Catalog public domain telecommunications disks provide the most inexpensive software you'll need. Be sure to read your on-disk documentation files carefully. Another outstanding value is the modem program on **HomePak** ($49.95, Batteries Included).

To enter the Data Library (DL) section of SIG*Atari, type DL# at a FUNCTION prompt (# is the desired section number from 0 to 6). Before downloading any file, check the filename extension. If it is .XMO, you must use a terminal program that supports XMODEM protocol such as AMODEM. To download a file with a .BIN extension you need a program that supports A-protocol such as TSCOPE. Both of these programs are found on Antic Catalog disk **PD024**. (If you have an **Atari 1030** or **835** modem you need disk **PD025**.)

Other extenders you'll run into here, such as .ATR or .CRE, are usually text files with LISTed BASIC programs that you can get with your capture buffer. Also, as you look around in the libraries, you'll find that many of the programs recommended below have associated files with a .DOC extender. These .DOC files typically contain the instruction documentation for the programs.

Another pattern you'll find in the libraries is that some of the larger programs are divided into multiple files, with numbers 1, 2, 3, etc. added to the central filename.

Now here is a representative sampling of some of the best SIG*Atari programs waiting for you to download. I only wish there was additional room in the magazine for me to list dozens more of the outstanding public domain programs that are yours on Compuserve. Also, I and **Antic** regret that space limitations forced the leaving out of authors' names in order to include the greatest possible number of programs.

DATA LIBRARY 1

This is the games library. And as you'll see, there's a real abundance of riches.

TRICKY.BIN: Good graphics and music combined with a novel game concept. You can play it as a strategy game and—if you're really quick—as an arcade game.

CASTLE.BIN: A great test of your mental prowess as you struggle through one of the most complex places on earth—Castle Hexagon.

CHICK.ATR: This screwball game never fails to reduce players and onlookers to rolling on the floor in laughter.

COUCH.ATR: Nothing like a session with the SIG's shrink. Read the inkblots, answer the probing questions and prepare yourself for the diagnosis.

BLKJCK.BIN: The premier blackjack game.

BEAMAT.BIN: Superb two-player duel of lasers, mirrors and wits. Written in compiled ACTION!

FIRWKS.CRE: Handel's "Music for the Royal Fireworks," complete with the fireworks display.

DOS4.BIN: The greatest Public Domain DOS. It won't format a disk or load a file, but its unique features have kept many a SIG*Atari member rolling on the floor.

FLIP.BIN: One of the best adaptations of the game of REVERSI. Many

interesting variations. Play against the computer or a friend.

DEDSTK.BIN: Fly SIG * Atari's space shuttle simulator on a dead stick landing.

NORAD.BIN: Terrify your friends with this simulated computer-tracked attack on the North American Air Defense Command. Only seconds separate you from doomsday!

IONPWR.BIN: Super sub-atomic pinball game.

POWER.ATR: Rule the World! Who could ask for anything more?

TREK.BIN (or .XMO): Deluxe version of the ever-popular game of STAR TREK—sound and color with its own on-board documentation.

LAUNCH.ATR: Skeet-shooting on Mars.

DATA LIBRARY 0

Here is the home of the SIG odds-n-ends—miscellaneous, unclassifiable, and sometimes totally weird.

DSKED.CRE: One of the neatest little sector editors around. Learn how information is stored on a disk and salvage the occasional ERROR 144 that crops up.

DATA LIBRARY 2

If SIG * Atari members have ONE thing in common, it's that they are all telecommunicators. And what ONE thing do all Telcommunicators need? Terminal programs, of course!

It only follows that the nation's largest group of Atari telecommunications activists would have the nation's largest supply of terminal programs! Here's a few that should turn on **Antic** readers.

TSCOPE.XMO: The program that started a revolution, still our number one reliable downloader. Prior to TSCOPE, our members who had machine language programs or programs using inverse video and control characters had to resort to heroic measures to upload their works.

Then TSCOPE implemented CompuServe's A-Protocol, providing error-free uploading and downloading from CIS. Our machine language programmers were free! But TSCOPE went even farther—it supported the 835 and 1030 modems as well! It even supported on-line VidTex cursor positioning and color graphics!

No MPP user on SIG * Atari can be without **TOPV35.CRE**. Unfortunately, the software that comes with the MPP does not permit file transfers to or from CompuServe with either XMODEM or "A" protocol. TOPV35.CRE cleverly solves that problem.

You want **AMODEM**? SIG * Atari has more versions of this popular terminal program than anybody—twenty-nine at last count!

Thinking about starting a BBS? The SIG also has plenty of public domain AMIS board programs available in DL 2.

DATA LIBRARY 3

This is where serious programmers love to hang out! In DL3 we have a complete tool box of utilities.

Want 80 columns on your screen? No problem! Check out **XDSPLY.BIN**. Nicely designed to fit right into your BASIC or machine language program.

You just deleted the wrong file? No problem again! Pick up **DOSWIZ.BIN** and recover that file immediately.

I put **MACH.DOS** on all my disks! It will run in double or single denity, reconfigure your drives, keep all your DOS utilities handy, and cause fewer memory conflicts than any other DOS available.

EXTBAS.CRE adds over 20 useful commands to Atari BASIC including RENUMBER, DELETE, and your most popular DOS utilities. This one even cross-references your variables!

My absolute favorite debugging utility is without question **LINE25.BIN**. This life-saver features a non-destructive trace that will find offending lines and data statements fast.

DATA LIBRARY 4

About six months ago DL 4 was opened up, a new Data Library dedicated to "Atari Arts—Sound and Graphics." The library's growth has exceeded the wildest expectations! Atari Arts is already the biggest library on the SIG!

MUSIC:

To enjoy the online music, you'll need an Atari Music System file player and SIG * Atari has the best! In DL 4 enter the command **BRO AMSXL.BIN** to retrieve a file player that is compatible with *all* Atari computers.

Now you've got over a hundred tunes to play, whatever your taste—jazz, classical, pop, country, Motown, Rhythm and Blues, Rock. Try your Atari's rendition of the Van Halen hit **JUMP.BIN**.

BACH.BIN is the entire Toccata and Fugue in D Minor.

Want Meredith Wilson? Try **TROM76.BIN**.

And here's a sample of other famous musical names you can download:

Michael Jackson — **THRILR.BIN**
Glenn Miller — **INMOOD.BIN**
Gershwin — **RAPBLU.BIN**
Scott Joplin — **ENTAIN.XMO**
The Doors — **LITFIR.BIN**
Ray Parker, Jr. — **BUSTER.BIN**
Beethoven — **MOONLT.BIN**

ART:

Painting with Light—the concept is mind-blowing in itself, but that's precisely what the Atari does better than any other computer! SIG * Atari members enjoy sharing their light-paintings in DL 4.

MAGICL.BIN is an outstanding graphics display program that will give you access to most of the online picture files, whether or not you have a touch tablet.

Supposedly you can't have more than four colors in ANTIC Mode E, but count the colors in **ODIE.BIN**—you'll see six!

Portraits—very difficult to do on a touch tablet. But look at **INDY1.BIN** for a sample of what can be created on the Atari.

And be sure to look at **MARS.BIN** for a fantastic other-world vista.

continued on next page

Don't miss **BUTTER.ACT** for some of the lushest colors you've ever seen. And **OCT.ATR** was beautiful enough to be on exhibit at Ohio State university.

There's also a fine selection of animated graphics demos that combine movement with color and form, to produce works of exquisite beauty.

DATA LIBRARY 5

The key to survival in this fast-changing environment is learning, and SIG*Atari has a whole section devoted to educational software in DL 5.

Want to learn assembly language? There's an entire on-line course devoted to Atari assembly programming. The file series starts with **AAC1.DOC**.

DL 5 is also where you find programs in Logo and ACTION! Just look around and you'll find lots to choose from.

There are also plenty of traditional educational programs in DL 5 that will be of great interest to parents, students and teachers.

Algebra students will want to try **GRAPHS.BIN**. It draws a high resolution graph of any two-variable equation expressible in BASIC. It can even solve simultaneous equations.

CATS.BIN allows teachers to compose, edit, administer, and score multiple choice and true-false tests right on their machines!

Of course there's much more, but this partial list gives you an idea of some of the best programs available for downloading from the CompuServe SIG*Atari. With this guide, you should be able to find your way around the SIG Data Libraries and find lots of other exciting and useful programs.

*Antic Contributing Editor Jerry White is a familiar PPN on SIG*Atari.*

SOFTWARE LIBRARY

Antic type-in listing section includes every full-length program from this issue. Listings are easier to type and proofread, easy to remove and save in a binder if you wish.

DISK SUBSCRIBERS: You can use all these programs immediately. Just follow the instructions in the accompanying magazine articles.

Antic program listings are typeset on the Star SG-10 printer—from Star Micronics, Inc., 200 Park Avenue, New York, NY 10166.

TYPING SPECIAL ATARI CHARACTERS

Antic printed program listings leave a small space between each Atari Special Character for easier reading. Immediately below you will see the way **Antic** prints all the standard Atari letters and numbers, in upper and lower case, in normal and inverse video.

ABCDEFGHIJKLMNOPQRSTUVWXYZ
ABCDEFGHIJKLMNOPQRSTUVWXYZ
abcdefghijklmnopqrstuvwxyz
abcdefghijklmnopqrstuvwxyz
0123456789 0123456789

The Atari Special Characters and the keys you must type in order to get them are shown in the two boxes below. (Squares are drawn around the normal video characters so you can see their positions more accurately, these squares will not appear in listings.)

NORMAL VIDEO

FOR THIS	TYPE THIS	FOR THIS	TYPE THIS
	CTRL ,		CTRL T
	CTRL A		CTRL U
	CTRL B		CTRL V
	CTRL C		CTRL W
	CTRL D		CTRL X
	CTRL E		CTRL Y
	CTRL F		CTRL Z
	CTRL G		ESC ESC
	CTRL H		ESC CTRL –
	CTRL I		ESC CTRL =
	CTRL J		ESC CTRL +
	CTRL K		ESC CTRL *
	CTRL L		CTRL .
	CTRL M		CTRL ;
	CTRL N	|	SHIFT =
	CTRL O		ESC SHIFT CLEAR
	CTRL P		ESC DELETE
	CTRL Q		ESC TAB
	CTRL R		
	CTRL S		

INVERSE VIDEO

FOR THIS	TYPE THIS	FOR THIS	TYPE THIS
	Λ CTRL ,		Λ CTRL Y
	Λ CTRL A		Λ CTRL Z
	Λ CTRL B		ESC SHIFT DELETE
	Λ CTRL C		ESC SHIFT INSERT
	Λ CTRL D		ESC CTRL TAB
	Λ CTRL E		ESC SHIFT TAB
	Λ CTRL F		Λ CTRL .
	Λ CTRL G		Λ CTRL ;
	Λ CTRL H		Λ SHIFT =
	Λ CTRL I		ESC CTRL 2
	Λ CTRL J		ESC CTRL DELETE
	Λ CTRL K		ESC CTRL INSERT
	Λ CTRL L		
	Λ CTRL M		
	Λ CTRL N		
	Λ CTRL O		
	Λ CTRL P		
	Λ CTRL Q		
	Λ CTRL R		
	Λ CTRL S		
	Λ CTRL T		
	Λ CTRL U		
	Λ CTRL V		
	Λ CTRL W		
	Λ CTRL X		

Whenever the CONTROL key (CTRL on the 400/800) or SHIFT key is used, *hold it down* while you press the next key. Whenever the ESC key is pressed, *release* it before you type the next key.

Turn on inverse video by pressing the Reverse Video Mode Key. Turn it off by pressing it a second time. (On the 400/800, use the Atari Logo Key instead.) Note: In the printed listings, inverse characters will be slightly smaller than the normal ones.

Among the most common program typing mistakes are switching certain capital letters with their lower-case counterparts—you need to look especially carefully at P, X, O and 0 (zero).

Some of Atari Special Characters are not easy to tell apart from standard alpha-numeric characters. Usually the Special Characters will be *thicker* than the alpha-numerics. Compare the two sets of characters below:

SPECIAL		STANDARD	
	CTRL F		/
	CTRL G		SHIFT +
	CTRL N		SHIFT -
	CTRL R		-
	CTRL S		+

HOW TO USE TYPO II

Type in TYPO II and SAVE a copy to disk or cassette.

Type GOTO 32000 and follow TYPO II onscreen instructions. If the resulting two-letter line codes are not exactly the same as those in the magazine, you mistyped something in that line.

To call back any line previously typed, type an asterisk [*] followed (without in-between spaces) by the line number, then press [RETURN]. When the complete line appears at the top of the screen, press [RETURN] again. This is also the way you use TYPO II to proofread itself.

To LIST your program, press [BREAK] and type LIST. To return to TYPO II, type GOTO 32000.

To remove TYPO II from your program, type LIST "D:FILENAME",0,31999 [RETURN] (Cassette owners LIST "C:). Type NEW, then ENTER "D:FILENAME" [RETURN] (Cassette—ENTER "C:). Your program is now in memory without TYPO II and you can SAVE or LIST it to disk or cassette.

Owners of the BASIC XL cartridge from O.S.S. type SET 5,0 and SET 12,0 before using TYPO II.

Don't type the TYPO II Codes!

```
WB 32000 REM TYPO II BY ANDY BARTON
VM 32010 REM VER. 1.0 FOR ANTIC MAGAZINE
HS 32020 CLR :DIM LINE$(120):CLOSE #2:CLO
   SE #3
BN 32030 OPEN #2,4,0,"E":OPEN #3,5,0,"E"
YC 32040 ? "↖":POSITION 11,1:? "TYPO II"
EM 32050 TRAP 32040:POSITION 2,3:? "Type
   in a program line"
HS 32060 POSITION 1,4:? " ":INPUT #2;LINE
   $:IF LINE$="" THEN POSITION 2,4:LIST B
   :GOTO 32060
XH 32070 IF LINE$(1,1)="*" THEN B=VAL(LIN
   E$(2,LEN(LINE$))):POSITION 2,4:LIST B:
   GOTO 32060
TH 32080 POSITION 2,10:? "CONT"
MF 32090 B=VAL(LINE$):POSITION 1,3:? " ";
NY 32100 POKE 842,13:STOP
CN 32110 POKE 842,12
```

Don't type the TYPO II Codes!

```
ET 32120 ? "↖":POSITION 11,1:? " TYPO II
   ":POSITION 2,15:LIST B
CE 32130 C=0:ANS=C
QR 32140 POSITION 2,16:INPUT #3;LINE$:IF
   LINE$="" THEN ? "LINE ";B;" DELETED":G
   OTO 32050
VV 32150 FOR D=1 TO LEN(LINE$):C=C+1:ANS=
   ANS+(C*ASC(LINE$(D,D))):NEXT D
WJ 32160 CODE=INT(ANS/676)
JW 32170 CODE=ANS-(CODE*676)
EH 32180 HCODE=INT(CODE/26)
BH 32190 LCODE=CODE-(HCODE*26)+65
HB 32200 HCODE=HCODE+65
IE 32210 POSITION 0,16:? CHR$(HCODE);CHR$
   (LCODE)
VG 32220 POSITION 2,13:? "If CODE does no
   t match press  RETURN  and edit line a
   bove.":GOTO 32050
```

ERROR FILE

MUSICIAN

June 1985

Change line 790 to:

```
790 IF A=54 THEN
POSITION 4,22:? #
6;"song cleared":
GOTO 810
```

MANEUVER

April 1985

If you get hearts on the title screen, LIST the program to disk or cassette, type NEW, then ENTER and SAVE it.

FONT MAKER FOR SG-10

March 1985

The July 1985 issue of ANTIC contains a listing which, when merged with FONT MAKER, makes that program work on the Star SG-10. See the HELP section of that issue for instructions.

KWIK DUMP

March 1985

The last number in line 1070 should be 27 instead of zero.

KOOKY'S QUEST

February '85

The following line is missing:

2100 FOR S=32 TO 16 STEP −4: SOUND 0,S,14,10: EA=EA *EA*EA: SOUND 0,0,0,0: EA=1 /\0:NEXT S

WIDE TEXT

January 1985

Substitute the following lines to print wide Z's.

In assembly:

```
0600       CPY #27*
8    ;8 BYTES TO A
 LETTER
```

In BASIC:

```
CT 20130 DATA 216
,208,213,185,0,22
4
```

ADVENT X-5

November '84

Missing line: 8020 RUN. Also, cassette owners should change the 138 in line 4005 to 130. The TYPO II code for line 1005 is EJ.

ADVENTURE ISLAND

November '84

Line 837 is missing its last item of data, a 4. Also, it will not run with DOS XL.

REVISION C CONVERTER

Article on page 25

LISTING 1

```
UJ 10 REM REV. B TO REV. C CONVERTER
VE 20 REM BY MATTHEW J.W. RATCLIFF
FW 30 REM (c) 1985, ANTIC PUBLISHING
AH 40 RESTORE
TB 50 GRAPHICS 0:DIM A$(10):? "        REVISION B TO C CONVERTER":? ,"FOR XL COMPUTERS":?
NI 60 ? "  This loader writes an AUTORUN.SYS":? "file to your disk.":?
VG 70 ? "  Please insert a DOS 2 disk":? "into Drive #1 and press [RETURN]."
DX 80 TRAP 140:INPUT A$
YO 90 CLOSE #1:OPEN #1,8,0,"D:AUTORUN.SYS"
OU 100 READ A:IF A<0 THEN 120
SZ 110 PUT #1,A:GOTO 100
XZ 120 CLOSE #1:? "** ALL DONE **"
RF 130 ? "SAVE THIS LOADER AS A BACKUP":? "JUST IN CASE!":END
OL 140 ? "ERROR # ";PEEK(195);" AT LINE ";PEEK(186)+256*PEEK(187):END
YS 1000 DATA 255,255,0,6,130,6,169,0,133,2
GQ 1010 DATA 169,6,133,3,173,250,3,240,1,96
SY 1020 DATA 169,0,133,216,169,160,133,217,160,0
JA 1030 DATA 173,1,211,41,253,141,1,211,177,216
IB 1040 DATA 72,173,1,211,9,2,141,1,211,104
HU 1050 DATA 145,216,230,216,208,228,230,217,165,217
FJ 1060 DATA 201,192,208,220,162,0,169,12,133,218
SK 1070 DATA 160,0,189,95,6,133,216,232,189,95
KD 1080 DATA 6,133,217,232,189,95,6,145,216,232
XE 1090 DATA 198,218,208,232,165,9,9,2,133,9
SY 1100 DATA 96,223,168,234,224,168,240,225,168,17
UC 1110 DATA 226,168,234,41,187,0,243,191,0,244
SN 1120 DATA 191,0,245,191,0,246,191,0,247,191
HU 1130 DATA 0,248,191,0,249,191,0,226,2,227
RU 1140 DATA 2,0,6,-1
```

LISTING 2

```
0250 ; REV.B TO REV.C CONVERTER
0260 ; BY MATTHEW RATCLIFF
0270 ; (c) 1985, ANTIC PUBLISHING
0280        *=  $0600
0290        .OPT OBJ
0300 ;
0310 ; PUT CODE INTO PAGE 6
0320 ;
0330 CASINI = $02       ;VECTOR WE WILL STEAL
0340 BOOT =   $09       ;CASS/DISK BOOT FLAG
0350 ; SET BIT 1 AND SYSTEM LOOKS FOR
0360 ; POST RESET HANDLER CODE POINTED
0370 ; TO BY CASINI VECTOR. THIS IS
0380 ; HOW RESET ALWAYS GIVES YOU REV.C
0390 ; RAM-BASIC.
0400 GINTLK = $03FA   ;EXTERNAL CART? DO NOTHING
0410 RMPNT = $D8      ;ROM/RAM POINTER
0420 COUNT = $DA      ;BUG FIX COUNT LOC
0430 PORTB = $D301    ;CONTROL ROM/RAM THRU HERE
0440 BEGBAS = $A000   ;BASIC ROM STARTS HERE
0450 ;
0460 ; STEAL CASSETTE RESET VECTOR
0470 ; AND POINT HERE FOR NEXT RESET.
0480 ;
0490 STEAL LDA # <STEAL
0500       STA CASINI   ;MUST RESTORE VECTOR
0510       LDA # >STEAL ;EVERY TIME RESET IS PRESSED
0520       STA CASINI+1
0530       LDA GINTLK   ;EXTERNAL CART?
0540       BEQ CNTNU    ;IF YES, CONTINUE
0550       RTS          ; IF NO, ABORT
0560 CNTNU LDA # <BEGBAS ;POINT TO START OF
0570       STA RMPNT    ;BAD BASIC ROM
0580       LDA # >BEGBAS
0590       STA RMPNT+1
0600 BCOPY LDY #0       ;COPY IT TO RAM
0610       LDA PORTB    ;BASIC ROM ON PLEASE
0620       AND #$FD     ;RESET CONTROL BIT=ON
0630       STA PORTB    ;GOT IT
0640       LDA (RMPNT),Y ;GET ROM BYTE
0650       PHA          ;SAVE IT
0660       LDA PORTB    ;NOW ENABLE THE RAM INSTEAD
0670       ORA #$02     ;SET CONTROL BIT=OFF
0680       STA PORTB    ;ROM OFF, RAM ON
0690       PLA          ;GET ROM BYTE
0700       STA (RMPNT),Y ;RAM VERSION UPDATED
0710       INC RMPNT
0720       BNE BCOPY
0730       INC RMPNT+1
0740       LDA RMPNT+1  ;DONE COPYING?
0750       CMP #$C0     ;END OF BASIC?
0760       BNE BCOPY    ;NOT YET!
0770 ; NOW GO FIX THE BUGS!
0780 FIXBUGS LDX #0     ;BUG TABLE INDEX
0790       LDA #12      ;12 BUGS TO FIX
0800       STA COUNT
0810 FXLOOP LDY #0      ;SETUP RAM POINTERS
0820       LDA FXTBL,X  ;GET ADDR LO BYTE
0830       STA RMPNT    ;SETUP RAM POINTER
0840       INX
0850       LDA FXTBL,X  ;GET ADDR HI BYTE
```

```
0860       STA RMPNT+1
0870       INX
0880       LDA FXTBL,X ;GET CORRECT BYTE

0890       STA (RMPNT),Y ;TO FIX THE BUG

0900       INX         ;POINT TO NEXT FI
X
0910       DEC COUNT
0920       BNE FXLOOP
0930       LDA BOOT    ;ALL DONE WITH FI
XED RAM-BASIC
0940       ORA #$02    ;RE-HOOK INTO BOO
T
0950       STA BOOT    ;FOR NEXT RESET K
EY-PRESS
0960       RTS
0970 ;
0980 ; ADDRESS,BYTE FIXUP TABLE FOR
0990 ; REV.B TO REV.C CONVERSION!
1000 ;
1010 FXTBL .WORD $A8DF ;ADDRESS
1020       .BYTE 234   ;WAS 202, NOW 234
 IN REV.C
1030       .WORD $A8E0
1040       .BYTE 240   ;WAS 208
1050       .WORD $A8E1
1060       .BYTE 17    ;WAS 10
1070       .WORD $A8E2
1080       .BYTE 234   ;WAS 96
1090 ;
1100 ; PEEK (43234)=96 FOR REV.B
1110 ; THAT IS ADDR $A8E2
1120 ;
1130       .WORD $BB29
1140       .BYTE 0     ;WAS 16
1150       .WORD $BFF3
1160       .BYTE 0     ;WAS 44
1170       .WORD $BFF4
1180       .BYTE 0     ;WAS 55
1190       .WORD $BFF5
1200       .BYTE 0     ;WAS 105
1210       .WORD $BFF6
1220       .BYTE 0     ;WAS 110
1230       .WORD $BFF7
1240       .BYTE 0     ;WAS 110
1250       .WORD $BFF8
1260       .BYTE 0     ;WAS 101
1270       .WORD $BFF9
1280       .BYTE 0     ;WAS 114
1290 ;
1300 ; ONLY 12 BYTES DIFFERENCE BETWEE
N
1310 ; REV.B & REV.C!
1320 ;
1330       .END
```

Antic's first 130XE program

ONE-PASS DISK COPY 130

Article on page 36

LISTING 1

```
02 ; COPY130
04 ; BY ERNIE NEGUS
06 ; (c) 1985, ANTIC PUBLISHING
10       *=  $2600    ;ROOM FOR DOS XL
20       JMP START
30 DBUFHI =   $0305    ;O.S. EQUATES
40 DBUFLO =   $0304    ;FOR OS I/O AND
50 DAUX2 =    $030B    ;DIRECT DISK I/O
60 DAUX1 =    $030A
70 DCOMND =   $0302
80 DUNIT =    $0301
90 ICCOM =    $0342
0100 ICBADR = $0344
0110 ICBLEN = $0348
0120 ICAUX1 = $034A
0130 ICAUX2 = $034B
0140 CARTR = $A000    ;MESSAGE AREA
0150 DES .BYTE "Destination #"
0160 FMT .BYTE "Format Destination ?"
0170 VER .BYTE "Verify Writes ?"
0180 STE .BYTE "Continue on Errors ?"
0190 RDG .BYTE $9B,"Reading..."
0200 WRT .BYTE $9B,"Writing..."
0210 IND .BYTE $9B,"Insert DESTINATION
 disk-Press Return"
0220 INS .BYTE "Insert SOURCE in Drive
 1-Type Return"
0230 ANC .BYTE $9B,$FD,"Another copy ?
"
0240 FRM .BYTE $9B,"Formatting..."
0250 REM .BYTE "Remove Cartridges Firs
t!",$9B
0260 O130 .BYTE "Program only runs on
a 130 XE!!",$9B
0270 TIT1 .BYTE $1D,"         SECTOR CO
PIER 130 XE   ",$1D,$9B
0280 TIT2 .BYTE "               By Ernie N
egus",$1D,$1D,$1D,$9B
0290 KEY .BYTE "K:"
0300 SCR .BYTE "S:"
0310 SECNT .BYTE 0     ;VARIABLES
0320 DEST .BYTE 0      ;USED BY PROGRAM
0330 FRMT .BYTE 0
0340 SOUR .BYTE 0
0350 DCOM .BYTE 0
0360 DTEM .BYTE 0
0370 SAVA .BYTE 0
0380 VERI .BYTE 0
0390 SPEC .BYTE 0
0400 SAVSPE .BYTE 0
0410 CERR .BYTE 0
0420 DRIVE .BYTE 0
0430 BPUT LDA #$26     ;OUTPUT TEXT
0440 BPUU STA ICBADR+1 ;SUBROUTINE
0450       LDA #0
0460       STA ICBLEN+1
0470       STX ICBLEN
0480       STY ICBADR
0490       LDX #0
0500       LDA #$0B
0510       STA ICCOM
0520       JSR $E456
0530       RTS
```

continued on next page

```
0540 BGET LDX #$10    ;INPUT BYTE FROM
0550      LDA #7      ;KEYBOARD SUB.
0560      STA ICCOM,X
0570      LDA #0
0580      STA ICBLEN,X
0590      STA ICBLEN+1,X
0600      JSR $E456
0610      STA $0600
0620      CMP #$1B    ;ESCAPE RERUNS
0630      BNE RET2    ;THE PROGRAM.
0640      PLA
0650      PLA
0660      JMP START
0670 RET2 RTS
0680 OPEN LDX #16     ;OPEN CHANNELS
0690      LDA #3      ;1 FOR KEYBOARD
0700      STA ICCOM,X ;6 FOR SCREEN.
0710      LDA # <KEY
0720      STA ICBADR,X
0730      LDA # >KEY
0740      STA ICBADR+1,X
0750      LDA #4
0760      STA ICAUX1,X
0770      JSR $E456
0780      LDX #$60
0790      LDA #3
0800      STA ICCOM,X
0810      LDA # <SCR
0820      STA ICBADR,X
0830      LDA # >SCR
0840      STA ICBADR+1,X
0850      LDA #12
0860      STA ICAUX1,X
0870      LDA #0
0880      STA ICAUX2,X
0890      JSR $E456
0900      RTS
0910 CLOSE LDX #$10   ;INSURES THOSE
0920      LDA #12     ;FILES CLOSED.
0930      STA ICCOM,X
0940      JSR $E456
0950      LDX #$60
0960      LDA #12
0970      STA ICCOM,X
0980      JSR $E456
0990      RTS
1000 START LDA CARTR ;THIS IS A
1010      INC CARTR   ;CARTRIDGE TEST
1020      CMP CARTR
1030      BNE NOCARTS
1040      LDX #25
1050      LDY # <REM
1060      JSR BPUT
1070      RTS
1080 NOCARTS LDA #$E3 ;TEST FOR 130XE
1090      STA 54017   ;BY SEEING IF
1100      STA $4000   ;THE BANK SWITCH
1110      LDA #255    ;WORKS.
1120      STA 54017
1130      LDA $4000
1140      CMP #$E3
1150      BNE ITSA130
1160      LDA #255
1170      STA 54017
1180      LDX #32
1190      LDY # <O130
1200      JSR BPUT
1210      RTS         ;RETURN TO DOS
1220 ITSA130 LDA #48
1230      STA 106     ;FORCE SCREEN
1240      JSR CLOSE   ;RAM BELOW $4000
1250      JSR OPEN
1260      JMP BEGIN
1270 PUTO LDA #$10    ;DRAWS THE
1280      STA 11546   ;DISPLAY WINDOW
1290      STA 11547   ;ON THE SCREEN.
1300      STA 11548
1310      LDA #128
1320      STA 11543
1330      STA 11545
1340      STA 11549
1350      STA 11551
1360      LDX #0
1370      STX 11550
1380 LOOX LDA #$55    ;DRAW BORDER.
1390      STA 11503,X
1400      LDA #$D5
1410      STA 11583,X
1420      INX
1430      CPX #9
1440      BNE LOOX
1450      RTS
1460 ECHO STA $0600   ;ECHO KEY TO
1470      LDA #$9B    ;SCREEN. DOES
1480      STA $0601   ;A LINE FEED.
1490      LDX #2
1500      LDY #0
1510      LDA #6
1520      JSR BPUU
1530      RTS
1540 BEGIN LDX #63    ;HERE IF 130 XE
1550      LDY # <TIT1
1560      LDA # >TIT1
1570      JSR BPUU    ;PRINT TITLE
1580      LDA #0
1590      STA SPEC
1600      LDA #148
1610      STA 710
1620      STA 712
1630      LDA #12
1640      STA 709
1650      LDX #13
1660      LDY # <DES
1670      JSR BPUT
1680 LAB2 JSR BGET    ;GET DEST. #
1690      CMP #'1     ;INSURE PROPER
1700      BCC LAB2    ;RANGE...
1710      CMP #'D     ;DUAL DRIVE
1720      BEQ DUAL
1730      CMP #'T     ;TRIPLE DRIVE
1740      BEQ TRIP
1750      CMP #'Q
1760      BEQ QUAD    ;QUAD DRIVE
1770      CMP #'9     ;UPPER RANGE.
1780      BCS LAB2
1790      JMP SING
1800 QUAD LDA #4      ;USED FOR
1810      BNE MULT    ;LOOPING WHEN
1820 TRIP LDA #3      ;MULTI-DEST.
1830      BNE MULT    ;ENABLED.
1840 DUAL LDA #2
1850 MULT STA SPEC
1860      STA SAVSPE  ;SAVE FOR
1870      LDA #'1     ;ANOTHER COPY.
1880 SING STA DEST    ;USE DRIVE 1
1890      STA DRIVE   ;IF MULTI-DEST.
1900      JSR ECHO
1910      LDA DEST
1920      SEC
1930      SBC #'0
1940      STA DEST
1950      LDX #20
1960      LDY # <FMT
1970      JSR BPUT
1980      JSR BGET    ;FORMAT OR NOT?
1990      JSR ECHO
2000      LDA $0600
2010      STA FRMT    ;SAVE ANSWER.
2020      LDX #15
2030      LDY # <VER
2040      JSR BPUT
2050      JSR BGET    ;VERIFY OR NOT?
2060      JSR ECHO
2070      LDA $0600
2080      STA VERI    ;SAVE ANSWER.
2090      LDA #255
2100      STA 54017
2110      LDX #20
2120      LDY # <STE  ;STOP ON
```

```
2130       JSR BPUT     ;ERROR?
2140       JSR BGET
2150       JSR ECHO
2160       LDA $0600
2170       STA CERR     ;SAVE ANSWER
2180       LDX #36
2190       LDY # <INS
2200       JSR BPUT
2210       JSR BGET
2220       LDA $0600    ;IF 'N' INSTEAD
2230       CMP #'N      ;OF RETURN, SKIP
2240       BEQ BLOOP    ;THE READ.
2250       LDX #11
2260       LDY # <RDG
2270       JSR BPUT
2280       JSR PUTO     ;PUT DISPLAY
2290       LDA #1       ;WINDOW UP AND
2300       STA DTEM     ;READ DATA OFF
2310       LDA #0       ;OF SOURCE.
2320       STA DAUX1
2330       STA DAUX2
2340       LDA #'R      ;'R' FOR READ
2350       STA DCOM
2360       LDA #255
2370       STA SECNT    ;READ IN 255
2380       JSR AVOID    ;SECTORS.
2390       LDA #$E3     ;ENABLE BANK 1
2400       STA 54017
2410       JSR DODISK
2420       LDA #$E7     ;ENABLE BANK 2
2430       STA 54017
2440       JSR DODISK
2450       LDA #$EB     ;ENABLE BANK 3
2460       STA 54017
2470       JSR DODISK
2480       LDA #$EF     ;ENABLE BANK 4
2490       STA 54017
2500       LDA #84      ;THIS TIME DO
2510       STA SECNT    ;NOT USE DEFAULT
2520       JSR AVOID    ;SECTOR COUNT.
2530 BLOOP LDA #255     ;THIS IS THE
2540       STA 54017    ;WRITE LOOP.
2550       LDA DEST
2560       CMP #1
2570       BNE AUTO
2580       LDX #37      ;INSERT PROMPT.
2590       LDY # <IND
2600       JSR BPUT
2610       JSR BGET
2620 AUTO LDA FRMT
2630       CMP #'Y      ;DID THEY WANT
2640       BNE SKYP     ;TO FORMAT?
2650       LDX #14
2660       LDY # <FRM
2670       JSR BPUT
2680       LDA #'!
2690       STA DCOMND   ;SET UP FORMAT
2700       LDA DEST
2710       STA DUNIT
2720       JSR PUTO
2730       JSR GONE     ;DISPLAY DRIVE #
2740       LDA #0
2750       STA DBUFLO
2760       LDA #4
2770       STA DBUFHI
2780       JSR $E453
2790       LDY #173
2800       LDA $0308
2810       BEQ SKYP
2820       LDX #'N
2830       STX CERR
2840       JSR ERRO     ;REPORT ERRORS
2850 SKYP LDX #11
2860       LDY # <WRT
2870       JSR BPUT
2880       JSR PUTO     ;DRAW WINDOW.
2890       LDA DEST     ;BEGIN WRITING.
2900       STA DTEM
2910       LDA #0
2920       STA DAUX2
2930       STA DAUX1
2940       LDA #'W      ;WRITE VERIFY?
2950       STA DCOM
2960       LDA VERI     ;<THE ANSWER.
2970       CMP #'Y
2980       BEQ SKIZZ
2990       LDA #'P      ;OR JUST PUT?
3000       STA DCOM
3010 SKIZZ LDA #255
3020       STA SECNT    ;WRITE OWN RAM.
3030       JSR AVOID    ;255 SECTORS.
3040       LDA #$E3     ;WRITE BANK 1.
3050       STA 54017
3060       JSR DODISK
3070       LDA #$E7     ;WRITE BANK 2
3080       STA 54017
3090       JSR DODISK
3100       LDA #$EB     ;WRITE BANK 3
3110       STA 54017
3120       JSR DODISK
3130       LDA #$EF     ;WRITE BANK 4
3140       STA 54017
3150       LDA #84      ;BUT DON'T USE
3160       STA SECNT    ;DEFAULT THIS
3170       JSR AVOID    ;TIME!
3180       LDA #255
3190       STA 54017    ;DISABLES BANKS
3200       LDA SPEC
3210       BEQ OVER     ;MULTI-DEST?
3220       CMP #1
3230       BEQ OVER     ;IF SO, DONE?
3240       STA DEST
3250       CLC
3260       ADC #'0      ;DO NEXT DRIVE
3270       STA DRIVE
3280       DEC SPEC
3290       JMP BLOOP    ;UNTIL FINISHED.
3300 OVER LDX #16
3310       LDY # <ANC
3320       JSR BPUT
3330       JSR BGET     ;ANOTHER COPY?
3340       JSR ECHO
3350       LDA $0600
3360       CMP #'Y
3370       BNE QUIT
3380       LDA SPEC     ;YES- RESTORE
3390       BEQ JUMP     ;VALUES IF MULTI
3400       LDA SAVSPE   ;DESTINATION
3410       STA SPEC     ;DRIVES.
3420       LDA #1
3430       STA DEST
3440       LDA #'1
3450       STA DRIVE    ;(FOR FORMAT).
3460 JUMP JMP BLOOP
3470 QUIT JMP START     ;RE-RUN PROGRAM
3480 DODISK LDA #127
3490       STA SECNT    ;THIS SUBROUTINE
3500 AVOID LDA #$3F     ;DOES THE DISK
3510       STA DBUFHI   ;I/O AND KEEPS
3520       LDA #$80     ;TRACK OF THE
3530       STA DBUFLO   ;MEMORY POINTERS
3540 DOSUP LDA DTEM
3550       STA DUNIT
3560       LDA DCOM
3570       STA DCOMND
3580 LOOP CLC
3590       LDA DAUX1
3600       ADC #1
3610       STA DAUX1
3620       LDA DAUX2
3630       ADC #0
3640       STA DAUX2
3650       CLC
3660       JSR INCO     ;IT ALSO KEEPS
3670       LDA DBUFLO   ;TRACK OF THE
3680       ADC #$80     ;SECTOR COUNTER.
3690       STA DBUFLO
3700       LDA DBUFHI
3710       ADC #0
```

continued on next page

```
3720      STA DBUFHI
3730      JSR $E453    ;THE O.S. CALL.
3740      CPY #1
3750      BEQ GDIO     ;IF ERROR FOUND
3760 ERRO JSR RER      ;DECODE IT AND
3770      STX 11550    ;DISPLAY IT IN
3780      LDA CERR
3790      CMP #'Y      ;STOP ON ERROR
3800      BEQ GDIO     ;UNLESS TOLD NOT
3810      PLA          ;TO.
3820      PLA
3830      JMP OVER
3840 GDIO DEC SECNT    ;THE RIGHT PART
3850      BNE LOOP     ;OF THE DISPLAY
3860      RTS          ;WINDOW.
3870 INCO LDX #3
3880 LOCO INC 11545,X
3890      LDA 11545,X  ;INCREMENTS THE
3900      CMP #$1A     ;SECTOR COUNTER
3910      BNE GONE     ;IN THE WINDOW.
3920      LDA #$10
3930      STA 11545,X
3940      DEX
3950      BNE LOCO
3960 GONE LDA DUNIT    ;ALSO DISPLAYS
3970      CLC          ;THE DRIVE
3980      ADC #$10     ;CURRENTLY IN
3990      STA 11544    ;USE.
4000      RTS
4010 RER  LDX #$10     ;DECODE ERRORS
4020      CPY #127     ;NON-IO ERROR
4030      BCC RETURN
4040      INX
4050      CPY #128     ;BREAK KEY ABORT
4060      BEQ RETURN
4070      INX
4080      CPY #138     ;DEVICE TIMEOUT
4090      BEQ RETURN
4100      INX
4110      CPY #139     ;NO ACKNOWLEDGE
4120      BEQ RETURN
4130      INX
4140      CPY #140     ;SERIAL BUS DATA
4150      BEQ RETURN
4160      CPY #142     ;ERRORS.
4170      BEQ RETURN
4180      CPY #143     ;DITTO
4190      BEQ RETURN
4200      INX
4210      CPY #144     ;BAD SECTOR/
4220      BEQ RETURN   ;OPEN DOOR.
4230      INX
4240      CPY #145     ;BAD READ VERIFY
4250      BEQ RETURN
4260      INX
4270      CPY #163     ;DAMAGED DOS
4280      BEQ RETURN
4290      INX
4300      CPY #164     ;BAD FILE NUMBER
4310      BEQ RETURN
4320      INX
4330      CPY #173     ;BAD SECTOR
4340      BEQ RETURN   ;DURING FORMAT.
4350      LDX #$10     ;NON-HANDLED ERRO
R
4360 RETURN RTS
4365 ;
4370      *=  $02E0    ;  FOR DOS 2
4380      .WORD START
4390      .END
```

LISTING 2

```
MJ 10 REM COPY 130
OY 20 REM BY ERNIE NEGUS
FW 30 REM (c) 1985, ANTIC PUBLISHING
MF 40 DIM FN$(20),TEMP$(20),AR$(93)
TX 50 ? "Output filename";:INPUT FN$
UK 70 IF FN$(1,1)="D" AND (FN$(2,2)=":" O
   R FN$(3,3)=":") THEN 90
MH 80 TEMP$(1,2)="D:":TEMP$(3)=FN$:FN$=TE
   MP$
NM 90 TRAP 180
PX 100 ? :? :? "Working...please stand by
   "
LO 110 RESTORE :READ LN:LM=LN:DIM A$(LN):
   C=1
BI 120 AR$="":READ AR$
XU 130 FOR X=1 TO LEN(AR$) STEP 3:POKE 75
   2,255
DE 140 LM=LM-1:POSITION 10,10:? "(Countdo
   wn...T-";INT(LM/10);")     "
SU 150 A$(C,C)=CHR$(VAL(AR$(X,X+2))):C=C+
   1:NEXT X:GOTO 120
IO 180 ? :? "Press START to write ";FN$
   ;"■";
BK 190 IF PEEK(53279)<>6 THEN 190
QO 200 OPEN #1,8,0,FN$:? #1;A$;:CLOSE #1
RU 220 GRAPHICS 0:? "COMPLETED"
HW 1000 DATA 1306
RY 1010 DATA 255255000038251038076211039O
   68101115116105110097116105111110032035
   070111114109097116032068101
XX 1020 DATA 115116105110097116105111100
   320630861011141051021210320871141051 16
   101115032063067111110116105
SU 1030 DATA 1101171010321111100320691141
   141111114115032063155082101097100105110
   103046046046155087114105116
ZV 1040 DATA 1051101030460460461550731101
   151011141160320680690830840730780650 84
   073079078032100105115107045
AB 1050 DATA 0801141011151150322102292442
   452422380731101151011141160320830790 85
   082067069032105110032068114
KC 1060 DATA 1051181010320490450841211121
   010322102292442452422381552530651101 11
   116104101114032099111112121
QI 1070 DATA 0320631550701111141090971161
   161051101030460460460821011091111181 01
   032067097114116114105100103
AL 1080 DATA 1011150320701051141151160331
   550801141111031140971090321111101081 21
   032114117110115032111110032
TN 1090 DATA 0970320490510480320880690330
   332520382470391550290320320320321 60
   160160211197195212207210160
GV 1100 DATA 1952072082011972101601771791
   761602161971601601600291550320320 32
   032032032032032032032066121
LV 1110 DATA 032069114110101051010320781 011
   031171150290290291550750580830580 00000
   000000000000000000000000000
MU 1120 DATA 0001690381410690031690001410
   730031420720031400680031620001690111 41
   066003032086228096162016169
GV 1130 DATA 0071570660031690001570720031
   570730030320862281410000062010272080 05
   104104076211039096162016169
AJ 1140 DATA 0031570660031690601570680031
   690391570690031690041570740030320862 28
   162096169003157066003169062
HJ 1150 DATA 1570680031690391570690031690
   121570740031690001570750030320862280 96
   162016169012157066003032086
RN 1160 DATA 2281620961690121570660030320
   862280961730001602380001602050001602 08
   008162025160196032076039096
RI 1170 DATA 1692271410012111410000641692
   551410012111730000642012272480392430 40
   208013169255141001211162032
SP 1180 DATA 1602210320760390961690481331
   060321900390321340390760840401690161 41
   026045141027045141028045169
XN 1190 DATA 1281410230451410250451410290
   451410310451620001420300451690851572 39
   044169213157063045232224009
```

```
VW 1200 DATA 2082410961410000061691551410
     01006162002160000169006032078039096162
     063160253169038032078039169
PP 1210 DATA 0001410720391691481411980021
     41200002169012141197002162013160003032
     076039032103039201049144249
EO 1220 DATA 2010682400232010842400152010
     81240007201057176233076162040169004208
     006169003208002169002141072
LH 1230 DATA 0391410730391690491410650391
     41075039032066040173065039056233048141
     065039162020160016032076039
EP 1240 DATA 0321030390320660401730000061
     41066039162015160036032076039032103039
     032066040173000006141071039
NA 1250 DATA 1692551410012111620201600510
     32076039032103039032066040173000006141
     074039162036244040239041160
UV 1260 DATA 1300320760390321030391730000
     06201078240073162011160071032076039032
     020040169001141069039169000
BB 1270 DATA 1410100031410110031690821410
     68039169255141064039032068042169227141
     001211032063042169231141001
DO 1280 DATA 2110320630421692351410012110
     32063042169239141001211169084141064039
     032068042169255141001211173
XV 1290 DATA 0650392010012080101620371600
     93032076039032103039173066039201089208
     052162014160182032076039169
NO 1300 DATA 0331410020031730650391410010
     03032020040032178042169000141004003169
     004141005003032083228160173
PW 1310 DATA 1730080032400081620781420740
     39032134042162011160082032076039032020
     040173065039141069039169000
SL 1320 DATA 1410110031410100031690871410
     68039173071039201089240005169080141068
     039169255141064039032068042
PR 1330 DATA 1692271410012110320630421692
     31141001211032063042169235141001211032
     063042169239141001211169084
DT 1340 DATA 1410640392400412350420320680
     42169255141001211173072039240019201001
     240015141065039024105048141
UI 1350 DATA 0750392060720390760760411620
     16160166032076039032103039032066040173
     000006201089208024173072039
WI 1360 DATA 2400161730730391410720391690
     01141065039169049141075039076076041076
     211039169127141064039169063
LW 1370 DATA 1410050031691281410040031730
     69039141001003173068039141002003024173
     010003105001141010003173011
AP 1380 DATA 0031050001410110030240321580
     42173004003105128141004003173005003105
     000141005003032083228192001
CH 1390 DATA 2400180321880421420300451730
     74039201089240005104104076016042206064
     039208189096162003254025045
OQ 1400 DATA 1890250452010262080081690161
     57025045202208238173001003024105016141
     024045096162016192127144055
RI 1410 DATA 2321921282400502321921382400
     45232192139240040232192140240035192142
     240031192143240027232192144
DR 1420 DATA 2400222321921452400172321921
     63240236042249042012232192164240007232
     192173240002162016096224002
IZ 1430 DATA 225002211039
```

machine language timing power for your programs

ATARI TIME MACHINE

Article on page 30

LISTING 1

Don't type the TYPO II Codes!

```
XM 2 REM TIMERS, LISTING 1
DI 4 REM BY FRED PINHO
FG 6 REM (c) 1985, ANTIC PUBLISHING
SC 10 GRAPHICS 1:DIM SEC$(3):POKE 710,0
AP 20 GOSUB 120:POSITION 4,7:? #6;"COUNT
   DOWN":? #6:? #6:? #6;"     TIMER DEMO"
QL 30 ? "TOTAL SECONDS DESIRED(1-250)";:I
   NPUT SEC$:SEC=VAL(SEC$):IF SEC>250 OR
   SEC<1 THEN ? "TRY AGAIN!":GOTO 30
NV 35 ? "TIMER DISPLAYS MINUTES:SECONDS"
XT 40 JFY=SEC*60:JFYHI=INT(JFY/256):JFYLO
   =JFY-(JFYHI*256):POKE 1025,JFYLO:POKE
   1026,JFYHI:POKE 1021,0
XR 50 TXTWD=PEEK(660)+256*PEEK(661):TMR=T
   XTWD+155:TMRHI=INT(TMR/256):TMRLO=TMR-
   (TMRHI*256):POKE 208,TMRLO
GO 60 POKE 209,TMRHI:POKE 752,1
WY 70 POKE 54286,0:POKE 548,0:POKE 549,6:
   ? "TIMER STARTED";:POKE 54286,64
FY 80 FOR T=1 TO 7:NEXT T
NT 90 A=PEEK(558):IF A THEN 90
ZM 100 ? "        TIMED OUT!";
YQ 110 FOR X=1 TO 750:NEXT X:RUN
TR 120 RESTORE 150
IW 130 FOR I=1536 TO 1703:READ Z:POKE I,Z
   :NEXT I
ZF 140 RETURN
JD 150 DATA 216,173,253,3,208,24,169,1,14
   1,4,4,169,255,141,253,3,141,46,2,172
WE 160 DATA 1,4,174,2,4,169,5,32,92,228,2
   06,4,4,208,117,169,10,141,4,4
GO 170 DATA 173,32,2,141,254,3,173,33,2,1
   62,8,14,254,3,42,201,60,144,5,233
HW 180 DATA 60,238,254,3,202,208,240,141,
   255,3,162,8,169,0,14,254,3,42,201,60
EO 190 DATA 144,5,233,60,238,254,3,202,20
   8,240,141,255,3,162,8,169,0,14,255,3
CM 200 DATA 42,201,10,144,5,233,10,238,25
   5,3,202,208,240,141,0,4,24,160,0,173
PR 210 DATA 254,3,105,16,145,208,200,169,
   26,145,208,200,173,255,3,105,16,145,20
   8,200
VQ 215 DATA 173,0,4,105,16,145,208,173,46
   ,2,240,3,76,98,228,169,98,141,36,2
JY 220 DATA 169,228,141,37,2,76,98,228
```

LISTING 2

```
0100 ;TIMER 2
0110 ;BY FRED PINHO
0120 ;(c) 1985, ANTIC PUBLISHING
0160        *=  $0600     ;RELOCATABLE
```

continued on next page

```
0170 ;SINCE MAC/65 USES PAGE 6, ASSEMB
LE TO DISK. RECOVER FILE WITH DOS.
0180 ;FOR DEBUGGING, ASSEMBLE WITH AN
OFFSET VIA THE .SET AND .OPT DIRECTIVE
S.
0190        .TITLE "LISTING 2"
0200        .TAB 14,18,26
0210        .SET 1,4
0220 SETVBV = $E45C
0230 XITVBV = $E462
0240        CLD             ;MUST BE DONE
0250        LDA 1021        ;TIMER ACTIVATED?

0260        BNE GOON        ;YES. SKIP AROUND

0270        LDA #1          ;NO. SET DELAY FO
R IMMEDIATE DISPLAY
0280        STA 1028
0290        LDA #255        ;SET TO INDICATE
ACTIVE TIMER
0300        STA 1021
0310        STA 558         ;SET TIMER FLAG
0320        LDY 1025        ;DESIRED TIME,LOW

0330        LDX 1026        ;HIGH
0340        LDA #5          ;TIMER 5 TO BE EN
ABLED
0350        JSR SETVBV      ;DOES THE JOB
0360 GOON DEC 1028      ;CALCULATE TIME?
0370        BNE EXIT        ;NOT YET. BYPASS
ROUTINE
0380        LDA #10         ;YES. RESET DELAY
 COUNTER
0390        STA 1028
0400        LDA 544         ;GET TIMER LO BYT
E
0410        STA 1022
0420        LDA 545         ;HI BYTE
0430 DIVIDE LDX #8      ;DIVIDE BY 60
0440 LOOP1 ASL 1022     ;TO GET TOTAL SEC
ONDS
0450        ROL A
0460        CMP #60
0470        BCC BRCH1
0480        SBC #60
0490        INC 1022
0500 BRCH1 DEX
0510        BNE LOOP1
0520        STA 1023
0530 ;TOTAL SECONDS STORED IN REGISTER
 1022
0540        LDX #8          ;DIVIDE BY 60
0550        LDA #0          ;TO GET MINUTES
0560 LOOP2 ASL 1022
0570        ROL A
0580        CMP #60
0590        BCC BRCH2
0600        SBC #60
0610        INC 1022
0620 BRCH2 DEX
0630        BNE LOOP2
0640        STA 1023
0650 ;MINUTES IN 1022,REMAINING SECOND
S IN 1023
0660 ;NOW DIVIDE BY 10 TO GET HI AND L
O DIGITS OF DECIMAL NUMBER
0670        LDX #8
0680        LDA #0
0690 LOOP3 ASL 1023
0700        ROL A
0710        CMP #10
0720        BCC BRCH3
0730        SBC #10
0740        INC 1023
0750 BRCH3 DEX
0760        BNE LOOP3

0770        STA 1024
0780 ;SECONDS(TENS PLACE) IN 1023
0790 ;SECONDS(UNIT PLACE) IN 1024
0800 DISPLAY CLC        ;STORE 3 DIGITS I
NTO DISPLAY AREA
0810        LDY #0
0820        LDA 1022        ;MINUTES
0830        ADC #16         ;CONVERT TO SCREE
N CODE
0840        STA (208),Y ;STORE
0850        INY
0860        LDA #26         ;SCREEN CODE FOR
':'
0870        STA (208),Y
0880        INY
0890        LDA 1023        ;SECONDS(HI)
0900        ADC #16
0910        STA (208),Y
0920        INY
0930        LDA 1024        ;SECONDS(LO)
0940        ADC #16
0950        STA (208),Y
0960        LDA 558         ;TIMER DONE?
0970        BEQ DISABLE ;YES
0980 EXIT JMP XITVBV ;NO. JUST RETURN
TO VBI
0990 DISABLE LDA #$62 ;RESTORE ORIGINA
L VBI VECTOR
1000        STA 548
1010        LDA #$E4
1020        STA 549
1030        JMP XITVBV      ;EXIT
1040        .END
```

LISTING 3

```
ZG 2 REM TIMERS, LISTING 3
DI 4 REM BY FRED PINHO
FG 6 REM (c) 1985, ANTIC PUBLISHING
OF 10 GRAPHICS 7:DL=PEEK(560)+256*PEEK(56
   1):POKE DL+76,7:POKE 710,194
WJ 20 FOR X=0 TO 8:POKE DL+78+X,PEEK(DL+8
   5+X):NEXT X
CO 25 COLOR 2:PLOT 0,70:DRAWTO 159,70:PLO
   T 80,71:DRAWTO 159,71:PLOT 0,72:DRAWTO
    79,72
RM 30 DD=PEEK(88)+256*PEEK(89):DD2=DD+284
   0
ZL 40 TMR=DD2+7:TMRHI=INT(TMR/256):TMRLO=
   TMR-(TMRHI*256):POKE 208,TMRLO:POKE 20
   9,TMRHI
RB 50 GOSUB 100:POKE 1021,0:POKE 1027,0:P
   OKE 755,0
NX 55 ? "TIMER DISPLAYS MINUTES:SECONDS"
TZ 60 POKE 54286,0:POKE 548,0:POKE 549,6:
   POKE 54286,64:? "TIMING!";:COLOR 1
DU 70 DEG :FOR Q=0 TO 359 STEP 0.5:R=COS(
   4*Q):X=R*COS(Q):Y=R*SIN(Q)
YD 80 PLOT INT((X*35)+79),INT((Y*30)+34):
   NEXT Q
EP 90 POKE 1027,255:? "▫            TASK CO
   MPLETED!":END
SL 100 RESTORE 130
QX 110 FOR I=1536 TO 1695:READ Z:POKE I,Z
   :NEXT I
ZB 120 RETURN
QS 130 DATA 216,173,253,3,208,18,169,1,14
   1,4,4,169,0,133,18,133,19,133,20,169
QN 140 DATA 255,141,253,3,206,4,4,208,115
   ,169,10,141,4,4,165,20,141,254,3,165
WC 150 DATA 19,162,8,14,254,3,42,201,60,1
   44,5,233,60,238,254,3,202,208,240,141
WU 160 DATA 255,3,162,8,169,0,14,254,3,42
   ,201,60,144,5,233,60,238,254,3,202
BV 170 DATA 208,240,141,255,3,162,8,169,0
   ,14,255,3,42,201,10,144,5,233,10,238
OZ 180 DATA 255,3,202,208,240,141,0,4,24,
   160,0,173,254,3,105,16,145,208,200,169
```

```
XD 190 DATA 26,145,208,200,173,255,3,105,
   16,145,208,200,173,0,4,105,16,145,208,
   173
XE 200 DATA 3,4,208,3,76,98,228,169,98,14
   1,36,2,169,228,141,37,2,76,98,228
```

LISTING 4

```
0100 ;TIMER 4
0110 ;BY FRED PINHO
0120 ;(c) 1985, ANTIC PUBLISHING
0160        *=  $0600     ;RELOCATABLE
0170 ;SEE LISTING 2 FOR REMARKS ON PAG
E 6 ASSEMBLY
0180        .TITLE "LISTING 4"
0190        .TAB 14,18,26
0200        .SET 1,4
0210 XITVBV = $E462
0220        CLD             ;MUST BE DONE
0230        LDA 1021        ;TIMER ACTIVATED?

0240        BNE GOON        ;YES. SKIP AROUND

0250        LDA #1          ;NO. SET DELAY FO
R IMMEDIATE DISPLAY
0260        STA 1028
0270        LDA #0          ;SET LOCATIONS TO
 0
0280        STA 18
0290        STA 19
0300        STA 20
0310        LDA #255        ;SET TO INDICATE
ACTIVE TIMER
0320        STA 1021
0330 GOON DEC 1028      ;CALCULATE TIME?
0340        BNE EXIT        ;NO. BYPASS ROUTI
NE
0350        LDA #10         ;YES. RESET DELAY
 COUNTER
0360        STA 1028
0370        LDA 20          ;GET TIMER COUNT
0380        STA 1022        ;LO BYTE
0390        LDA 19          ;HI BYTE
0400 DIVIDE LDX #8      ;DIVIDE BY 60
0410 LOOP1 ASL 1022     ;TO GET TOTAL SEC
ONDS
0420        ROL A
0430        CMP #60
0440        BCC BRCH1
0450        SBC #60
0460        INC 1022
0470 BRCH1 DEX
0480        BNE LOOP1
0490        STA 1023
0500 ;TOTAL SECONDS STORED IN REGISTER
 1022
0510        LDX #8          ;DIVIDE BY 60
0520        LDA #0          ;TO GET MINUTES
0530 LOOP2 ASL 1022
0540        ROL A
0550        CMP #60
0560        BCC BRCH2
0570        SBC #60
0580        INC 1022
0590 BRCH2 DEX
0600        BNE LOOP2
0610        STA 1023
0620 ;MINUTES IN 1022, REMAINING SECON
DS IN 1023
0630 ;NOW DIVIDE BY 10 TO GET HI AND L
O DIGITS OF DECIMAL NUMBER
0640        LDX #8
0650        LDA #0
0660 LOOP3 ASL 1023
0670        ROL A
0680        CMP #10
0690        BCC BRCH3
0700        SBC #10
0710        INC 1023
0720 BRCH3 DEX
0730        BNE LOOP3
0740        STA 1024
0750 ;SECONDS(TENS PLACE) IN 1023
0760 ;SECONDS(UNIT PLACE) IN 1024
0770 DISPLAY CLC        ;STORE 3 DIGITS I
NTO DISPLAY AREA
0780        LDY #0
0790        LDA 1022        ;MINUTES
0800        ADC #16         ;CONVERT TO SCREE
N CODE
0810        STA (208),Y ;STORE
0820        INY
0830        LDA #26         ;SCREEN CODE FOR
':'
0840        STA (208),Y
0850        INY
0860        LDA 1023        ;SECONDS(HI)
0870        ADC #16
0880        STA (208),Y
0890        INY
0900        LDA 1024        ;SECONDS(LO)
0910        ADC #16
0920        STA (208),Y
0930        LDA 1027        ;TIMER DONE?
0940        BNE DISABLE ;YES
0950 EXIT JMP XITVBV ;NO. JUST RETURN
TO VBI
0960 DISABLE LDA #$62 ;RESTORE ORIGINA
L VBI VECTOR
0970        STA 548
0980        LDA #$E4
0990        STA 549
1000        JMP XITVBV   ;EXIT
1010        .END
```

LISTING 5

```
BA 2 REM TIMERS, LISTING 5
DI 4 REM BY FRED PINHO
FG 6 REM (c) 1985, ANTIC PUBLISHING
QF 10 GRAPHICS 7:DL=PEEK(560)+256*PEEK(56
   1):POKE DL+76,7:POKE 710,194
WJ 20 FOR X=0 TO 8:POKE DL+78+X,PEEK(DL+8
   5+X):NEXT X
CF 30 COLOR 2:PLOT 0,70:DRAWTO 159,70:PLO
   T 80,71:DRAWTO 159,71:PLOT 0,72:DRAWTO
    79,72
LR 40 DD=PEEK(88)+256*PEEK(89):DD2=DD+284
   0:POKE 18,0:POKE 19,0:POKE 20,0:POKE 7
   52,1
YX 50 ? "TIMER DISPLAYS MINUTES:SECONDS":
   ? "TIMING!";:COLOR 1:TMR=DD2+7:POKE TM
   R+1,26
OE 60 DEG :FOR Q=0 TO 359 STEP 0.5
DH 70 JIFF=PEEK(20)+256*PEEK(19)+65536*PE
   EK(18):SECS=INT(JIFF/60):MINS=INT(SECS
   /60):SECS=SECS-60*MINS
RF 80 TSECS=INT(SECS/10):SECS=SECS-10*TSE
   CS
KT 90 POKE TMR,MINS+16:POKE TMR+2,TSECS+1
   6:POKE TMR+3,SECS+16
KK 100 R=COS(4*Q):X=R*COS(Q):Y=R*SIN(Q)
US 110 PLOT INT((X*35)+79),INT((Y*30)+34)
   :NEXT Q
GZ 120 ? "↖          TASK COMPLETED!":END
```

eye-popping demonstration of "page flipping"

TIC TOC FLIP

Article on page 27

LISTING 1

Don't type the TYPO II Codes!

```
KJ 5 REM TIC-TOC-FLIP
EL 6 REM BY GENE LEVINE
FH 7 REM (c) 1985, ANTIC PUBLISHING
CH 10 POKE 106,PEEK(740):GOTO 2000
HL 19 REM ANIMATION LOOP
CT 20 FOR L=0 TO 87 STEP 8:POKE 561,H1-L:
   GOSUB 40:NEXT L:SOUND 0,25,10,14
JN 25 FOR L=0 TO 71 STEP 8:POKE 561,H11+8
   +L:GOSUB 40:NEXT L:SOUND 0,33,10,14
HI 30 POKE 77,0:GOTO 20
EH 39 REM SPEED SET & ESC
UX 40 SOUND 0,0,0,0:FOR DELAY=1 TO S:IF P
   EEK(764)<29 THEN GOSUB 50
BG 45 NEXT DELAY:RETURN
UT 50 IF PEEK(764)=14 AND S<70 THEN S=S+0
   .3:REM '-' TO SLOW DOWN
NQ 55 IF PEEK(764)=6 AND S>0 THEN S=S-0.2
   :REM '+' TO SPEED UP
YV 60 IF PEEK(764)=28 THEN 9999:REM ESC T
   O STOP
VK 65 POKE 764,255:RETURN
JP 99 REM SCREEN DRAW SETUP
LM 100 POKE 106,T-TX:GRAPHICS 5+16:POKE 5
   59,34:POKE 712,PEEK(53770):LFT=16:RIT=
   63:BOT=47:TOP=0:TX=TX+8
QB 199 REM DRAW BOX
SE 200 FOR L=1 TO 12:FOR K=1 TO 2:COLOR K
   :PLOT LFT,TOP:DRAWTO RIT,TOP:DRAWTO RI
   T,BOT:DRAWTO LFT,BOT:DRAWTO LFT,TOP
BP 205 BOT=BOT-1:TOP=TOP+1:LFT=LFT+X:RIT=
   RIT-XX:NEXT K:NEXT L:X=X+0.32:XX=XX-0.
   32
YB 299 REM DRAW TICK-TOCKER
ZW 300 COLOR 3:PLOT 38,V:DRAWTO 41,V:DRAW
   TO 41,VV:DRAWTO 38,VV:DRAWTO 38,V
GP 305 PLOT 39,V-1:DRAWTO 40,V-1:PLOT 39,
   VV+1:DRAWTO 40,VV+1
RY 310 V=V+4:VV=VV+4:RETURN
ZD 999 REM D.L. HIGH BYTES
WR 1000 GOSUB 100:COLOR 1:PLOT 39,0:DRAWT
   O 40,0:H1=PEEK(561)
OX 1005 FOR Q=2 TO 11:GOSUB 100:NEXT Q
SV 1010 COLOR 1:PLOT 39,47:DRAWTO 40,47:H
   11=PEEK(561)
HX 1499 REM SET COLORS & TURN ON SCREEN
HT 1500 POKE 712,0:POKE 708,146:POKE 709,
   66:POKE 710,214:POKE 559,34:GOTO 20
OE 1999 REM INITIALIZE VARIABLES
EY 2000 DH=PEEK(561):T=PEEK(106):TX=0:V=1
   :VV=6:X=-0.6:XX=2+0.6:S=10:GOTO 1000
FR 9998 REM EXIT
OK 9999 POKE 561,DH:POKE 106,T:GRAPHICS 0
   :REM RESET DL HIGH BYTE & MEM TOP
```

fast, unusual graphics in half the memory

MIRRORED DISPLAY LISTS

Article on page 33

LISTING 1

Don't type the TYPO II Codes!

```
EZ 100 REM REFLECT
XQ 200 REM BY DAVID PLOTKIN
IL 300 REM (c) 1985, ANTIC PUBLISHING
YG 800 POKE 106,PEEK(740)
IN 900 GRAPHICS 8+16:POKE 709,0:POKE 710,
   0
MP 1000 YTOP=PEEK(106)*256-4*1024:YSTART=
   YTOP:DLS=YSTART-512:MEMTOP=DLS-1
KB 1010 HIGH=INT(MEMTOP/256):LOW=MEMTOP-H
   IGH*256:POKE 741,LOW:POKE 742,HIGH:POK
   E DLS,112:POKE DLS+1,112
KJ 1020 POKE DLS+2,112:POKE DLS+3,79:HIGH
   =INT(YSTART/256):LOW=YSTART-HIGH*256:P
   OKE DLS+4,LOW:POKE DLS+5,HIGH
HI 1025 POKE 88,LOW:POKE 89,HIGH:FOR CNT=
   6 TO 100:POKE DLS+CNT,15:NEXT CNT:ADRS
   =YSTART+96*40
XD 1040 FOR CNT=101 TO 389 STEP 3:ADRSH=I
   NT(ADRS/256):ADRSL=ADRS-ADRSH*256
RF 1050 POKE DLS+CNT,79:POKE DLS+CNT+1,AD
   RSL:POKE DLS+CNT+2,ADRSH
BO 1060 ADRS=ADRS-40:NEXT CNT
FC 1070 POKE DLS+392,65:HIGH=INT(DLS/256)
   :LOW=DLS-HIGH*256:POKE DLS+393,LOW:POK
   E DLS+394,HIGH
PQ 1080 POKE 560,LOW:POKE 561,HIGH:POKE 7
   09,202:POKE 54272,34
OK 1090 PRINT #6;CHR$(125)
AM 1100 FOR X=5 TO 315 STEP 5:COLOR 1
PG 1110 FOR XX=0 TO 319 STEP 5:PLOT X,95:
   DRAWTO XX,0:NEXT XX
CW 1120 COLOR 0:FOR XX=0 TO 319 STEP 5:PL
   OT X-5,95:DRAWTO XX,0:NEXT XX
PP 1130 NEXT X:GOTO 1100
```

game of the month

CRICKETS

Article on page 13

LISTING 1

Don't type the TYPO II Codes!

```
LB 100 REM COURTING CRICKETS
VK 110 REM BY STAN OCKERS
KV 120 REM ANTIC PUBLISHING
FD 140 ? "↖INITIALIZING ....."
IV 150 DIM D$(1),F$((INT(ADR(D$)/2048)+1)
   *2048-ADR(D$)-1),DD$(1024),P0$(256),P1
   $(256),P2$(256),P3$(256)
LF 160 DIM BL$(13),CR$(12),CRJ$(12),CRF1$
   (12),CRF2$(12)
KR 161 DIM CRM$(12),STK$(32),HT1$(12),HT2
   $(12),SND$(173):HI=ADR(DD$)/256
GR 170 DIM GF1$(15),GF2$(15),DRP$(13),CUR
   $(12),COLDR$(4):GF1$="HIXLMXPQXTUXXXX"
   :GF2$="JKXNOXRSXVWXZ[X"
ID 180 REM JOYSTICK ROUTINE
KP 190 RESTORE 200:FOR J=1 TO 32:READ A:S
   TK$(J,J)=CHR$(A):NEXT J:STK=ADR(STK$)
HK 200 DATA 104,173,132,2,240,12,173,207,
   6,240,20,169,0,141,207,6,240,13,173,20
   7,6,208,8,173,120,2,41,3,141,207,6,96
XX 210 REM SOUND DATA
UR 220 RESTORE 230:FOR J=1 TO 173:READ A:
   SND$(J,J)=CHR$(A):NEXT J
EH 230 DATA 0,60,3,70,3,80,3,90,3,100,3,1
   00,3,0,0,108,10,108,10,108,10,81,20,64
   ,10,108,10,108,10,108,10,81,20
LI 240 DATA 64,10,0,30,53,10,85,10,53,10,
   96,10,53,10,96,10,53,10,91,20,102,10,1
   08,10,108,10,85,20,72,10,108,10
GQ 250 DATA 108,10,108,10,85,20,72,10,0,5
   0,53,15,47,5,53,10,60,10,64,10,72,10,8
   1,20,40,20,0,0
QF 260 DATA 121,40,91,30,91,10,91,80,121,
   40,81,30,96,10,91,80,121,40,91,30,72,1
   0,60,40,72,30,91,10,91,40,96,30
OM 270 DATA 91,10,81,80,0,0,243,40,243,40
   ,243,10,243,40,204,40,217,10,217,40
WY 280 DATA 243,10,243,40,255,10,243,60,0
   ,0,47,10,72,10,60,10,64,10,72,10,64,10
   ,60,10,72,10,0,0
JA 290 HS=INT(ADR(SND$)/256):POKE 209,HS:
   LS=ADR(SND$)-HS*256:POKE 208,LS:SOUND
   3,0,0,0
WQ 300 DIF=1
KZ 310 REM DD$ IS SCREEN DATA
XQ 320 DD$(1)="e":DD$(448)="e":DD$(2)=DD$
   :DD$(449)=CHR$(0):DD$(828)=CHR$(0):DD$
   (450)=DD$(449)
NN 330 LINE1=ADR(DD$)+513:HL=INT(LINE1/25
   6):LL=LINE1-HL*256:POKE 88,LL:POKE 89,
   HL
GI 340 POKE 559,0:POKE 106,PEEK(740):GOSU
   B 1140:GOSUB 1330:GOSUB 1530
LS 350 GRAPHICS 0:POKE 559,0:FOR X=53248
   TO 53250:POKE X,1:NEXT X
QF 355 POKE 756,START/256:GOSUB 1380:POKE
    559,0:POKE 560,0:POKE 561,6:POKE 559,
   34:GOSUB 1710
LZ 360 A=USR(1670):VERT=8:GOSUB 1780:POKE
    88,LL:POKE 89,HL:POSITION 12,0:? DIF:
   BRO=0
FX 370 POKE 708,68:POKE 709,254:POKE 710,
   86:POKE 711,44:POKE 712,72:POSITION 16
   ,7:? " courting crickets "
ZW 380 POSITION 2,0:? "dif level":RESTORE
    382:FOR J=1 TO 4:READ A:COLDR$(J,J)=C
   HR$(A):NEXT J
JR 382 DATA 228,36,4,230
MZ 390 DD$(613,652)="YYYYYYXXXXXYYYYYYXXX
   XXYYYYYYXXXXXYYYYYY"
QA 400 DD$(653,692)="YYYYYYXXXXXYYYYYYXXX
   XXYYYYYYXXXXXYYYYYY"
OH 410 DD$(693)="_◆XX_◆XX_◆":DD$(769)="ab
   XXabXXabX"
NN 420 GN=0:WFLG=0:GIFT=0:DROP=0:DD$(552)
   ="XXXXXXXXXXXXXX":DD$(592)="XXXXXXXXXX
   XXXX"
BC 430 POKE 1622,15:DD$(712)=GF1$:DD$(788
   )=GF2$
OM 440 POSITION 16,7:? "SELECT dif or FIR
   E":POKE 77,0
ZN 450 IF STRIG(0)=0 THEN 450
LC 460 KEY=PEEK(53279):IF STRIG(0)=0 THEN
    500
BY 470 IF KEY<>5 THEN 460
MM 480 DIF=DIF+1:IF DIF>7 THEN DIF=1
HY 490 POSITION 12,0:? DIF:GOSUB 1780:GOT
   O 460
JG 500 POSITION 16,7:? " COURTING CRICKET
   S ":GOSUB 1000:DD$(712)="XX":DD$(788)=
   "XX"
XN 510 A=USR(ADR(STK$))
IC 520 IF GIFT=1 AND YPOS=187 AND P>100 A
   ND P<134 THEN GOSUB 820:GOSUB 790
ZA 530 IF PEEK(1743)=0 THEN FLAG=0
XZ 540 IF FLAG=1 THEN 580
OP 550 S=PEEK(1743):IF S=2 OR S=1 THEN FL
   AG=1:P0$(YPOS)=CRJ$:POKE 53767,170:POK
   E 1591,1:FOR J=1 TO 20:NEXT J
CW 560 IF S=2 AND VERT>0 THEN POKE 1767+V
   ERT,0:P0$(YPOS)=BL$:YPOS=YPOS-16:P0$(Y
   POS)=CR$:VERT=VERT-1:POKE 1767+VERT,1
UY 570 IF S=1 AND VERT<8 THEN POKE 1767+V
   ERT,0:P0$(YPOS)=BL$:YPOS=YPOS+16:P0$(Y
   POS)=CR$:VERT=VERT+1:POKE 1767+VERT,1
ZQ 580 P=PEEK(1791):IF P>190 OR P<60 THEN
    POKE 1767+VERT,0:GOTO 710
UQ 590 POKE 53278,0
PZ 600 FOR J=1 TO 10:NEXT J
YA 610 IF PEEK(53252)>0 THEN POKE 1767+VE
   RT,0:GOTO 710
FH 620 IF YPOS=59 AND P>100 AND P<134 THE
   N POKE 1767+VERT,0:GOTO 840
OF 630 JPOS=JPOS+DELJ:IF JPOS>RTLJ OR JPO
   S<LLJ THEN DELJ=-DELJ:JPOS=JPOS+2*DELJ

QQ 640 POKE 53250,JPOS
JL 650 DRCNT=DRCNT-1:IF DRCNT<1 THEN DRCN
   T=10+5*(10-DIF):DPOS=50:GOSUB 950:P3$(
   DPOS)=DRP$:DROP=1:POKE 53251,JPOS
XL 660 IF DROP=1 THEN P3$(DPOS)=BL$:DPOS=
   DPOS+DELTA:P3$(DPOS)=DRP$:SOUND 1,DPOS
   -40,10,10
EF 670 IF DPOS>240 THEN GOSUB 820:DPOS=50

DH 680 IF PEEK(53260)=8 THEN POKE 1767+VE
   RT,0:GOTO 710
OC 690 GOTO 510
```

continued on next page

```
EX 700 REM FALLING CRICKET
UR 710 GOSUB 820
TY 720 P0$(YPOS)=BL$:YPOS=YPOS+6:P0$(YPOS
   )=CRF1$:SOUND 0,YPOS,10,10:FOR J=1 TO
   30:NEXT J
RH 730 P0$(YPOS)=BL$:YPOS=YPOS+6:P0$(YPOS
   )=CRF2$:SOUND 0,YPOS,10,10:FOR J=1 TO
   30:NEXT J:IF YPOS<240 THEN 720
VZ 740 SOUND 0,0,0,0
KY 750 BRO=BRO+1:IF BRO=4 THEN 1030
SN 760 J=4*(BRO-1):DD$(693+J,696+J)="XXXX
   ":DD$(769+J,772+J)="\]^X"
EC 770 VERT=8:GOSUB 1710:POKE 1791,120:PO
   KE 1622,15:GOTO 510
XU 780 REM ERASE NEXT GIFT
WV 790 GIFT=0:GN=GN+1:GOSUB 1000:DD$(712+
   GN*3)="XX":DD$(788+GN*3)="XX"
ZE 800 RETURN
BL 810 REM ELIMINATE DROP
MK 820 SOUND 1,0,0,0:P3$(DPOS)=BL$:POKE 5
   3251,0:DROP=0:RETURN
TV 830 REM REACHED FEMALE
LS 840 GOSUB 820:POKE 1791,118
DX 850 IF GIFT=0 THEN GOSUB 920
UE 860 IF WFLG=1 THEN 1090
WA 870 FOR K=1 TO 10:P1$(YPOS-14)=HT1$
IS 880 FOR J=15 TO 0 STEP -1:SOUND 0,20,1
   0,J:NEXT J:P1$(YPOS-14)=HT2$:FOR J=1 T
   O 15:NEXT J:NEXT K
IE 890 IF STRIG(0)=1 THEN 890
NS 900 P0$(YPOS)=BL$:P1$(YPOS-14)=BL$:S=1
   :GOTO 560
LW 910 REM PRINT GIFT
YO 920 GIFT=1:DD$(552)=GF1$(1,(GN+1)*3):D
   D$(592)=GF2$(1,(GN+1)*3):IF GN=4 THEN
   WFLG=1
ZL 930 RETURN
VA 940 REM PICK A WEAPON
BA 950 R=INT(RND(0)*4):RESTORE 960+10*R:F
   OR J=1 TO 13:READ A:DRP$(J,J)=CHR$(A):
   NEXT J:POKE 707,ASC(COLDR$(R+1))
ZV 952 RETURN
QQ 960 DATA 20,72,34,20,74,40,8,127,127,6
   2,62,28,28
OH 970 DATA 0,0,80,112,112,112,112,112,12
   0,126,94,0,0
LU 980 DATA 0,0,48,96,64,127,127,64,224,2
   24,0,0,0
ZN 990 DATA 60,24,24,24,60,126,223,215,24
   7,255,127,126,60
ZK 995 REM PICK ANOTHER GIFT
CS 1000 POKE 1622,155:POKE 707,92:POKE 53
   251,124+12*GN:FOR J=1 TO 5:P3$(205)=CU
   R$:FOR K=1 TO 30:NEXT K
PJ 1010 P3$(205)=BL$:FOR K=1 TO 30:NEXT K
   :NEXT J:RETURN
UU 1020 REM NO MORE BROTHERS
RV 1030 POKE 1622,131:POKE 53277,0:FOR J=
   53261 TO 53264:POKE J,0:NEXT J:GRAPHIC
   S 18:POSITION 4,3:? #6;"All Brothers"
CQ 1040 POSITION 6,4:? #6;"are Gone"
BS 1050 POSITION 3,7:? #6;"PRESS start TO
   ":POSITION 5,8:? #6;"try again"
UA 1060 IF PEEK(53279)<>6 THEN 1060
QR 1070 GOTO 350
AJ 1080 REM MARRAIGE OCCURS
KB 1090 POKE 1622,93:FOR L=0 TO 6:FOR K=0
    TO 3:POS=64*L+16*K:DD$(POS+1)="XXXXcd
   XXXXcdXXXX":NEXT K:NEXT L
ZW 1100 FOR J=1 TO 1000:NEXT J
WK 1110 POKE 53277,0:FOR J=53261 TO 53264
   :POKE J,0:NEXT J:GRAPHICS 18:POSITION
   3,3:? #6;"And They Lived"
YA 1120 POSITION 1,4:? #6;"Happily Ever A
   fter":GOTO 1050
SU 1130 REM CHANGE CHARACTER SET
TG 1140 DIM ZZ$(32):RESTORE 1150:FOR I=1
   TO 32:READ A:ZZ$(I)=CHR$(A):NEXT I
RT 1150 DATA 104,104,133,204,104,133,203,
   104,133,206,104,133,205,162,4,160,0
OO 1160 DATA 177,203,145,205,136,208,249,
   230,204,230,206,202,208,240,96
FZ 1170 POKE 106,PEEK(106)-5:START=(PEEK(
   106)+1)*256
YT 1180 A=USR(ADR(ZZ$),57344,START):RESTO
   RE 1200:FOR I=START+512 TO START+807:R
   EAD A:POKE I,A:NEXT I
BC 1190 RETURN
YN 1200 DATA 0,0,0,0,0,0,0,170,0,63,58,57
   ,58,63,63,170,0,252,172,108,172,252,25
   2,170
NX 1210 DATA 0,255,170,85,170,255,255,170
   ,0,51,33,18,33,255,85,170,0,191,239,25
   1,254,255,255,170
YG 1220 DATA 0,255,215,215,215,190,235,17
   0,0,254,251,239,191,255,255,170,0,1,32
   ,184,32,12,3,16
ES 1230 DATA 64,208,64,224,184,224,192,19
   2,116,28,3,0,0,0,0,0,200,238,248,192,1
   92,192,192,192
DM 1240 DATA 0,0,0,0,2,2,3,3,0,0,0,0,128,
   128,192,192,13,55,219,222,223,55,13,3
AU 1250 DATA 112,220,247,247,247,220,112,
   192,0,0,0,58,234,234,233,229,0,0,0,40,
   234,170,154,86
RT 1260 DATA 233,233,57,58,58,14,14,13,86
   ,90,90,104,104,160,160,128,0,3,12,48,1
   92,192,48,12
PY 1270 DATA 48,204,3,3,3,12,48,12,12,3,1
   2,48,14,2,2,0,3,3,3,12,32,160,160,128
SY 1280 DATA 0,0,0,0,0,0,0,0,233,181,173,
   183,222,122,94,107,2,10,2,4,16,16,4,1,
   128,160,128,16,4,4,16,64
HX 1290 DATA 0,0,3,15,15,204,63,15,0,0,0,
   207,204,255,255,195,0,0,240,48,48,48,2
   40,240
PG 1300 DATA 0,0,84,5,17,5,1,5,0,0,21,80,
   68,80,64,80,17,65,1,4,4,20,0,0
ZR 1310 DATA 68,65,64,16,16,20,0,0,40,190
   ,179,176,176,44,11,2,80,244,52,52,52,2
   08,64,0
TR 1320 REM VBI ROUTINE
JY 1330 DIM VBI$(75):RESTORE 1340:FOR J=1
    TO 75:READ A:VBI$(J,J)=CHR$(A):NEXT J
   :VBI=ADR(VBI$):RETURN
UF 1340 DATA 216,162,0,160,0,222,240,6,16
   ,42,189,224,6,157,240,6,189,232,6,240,
   10,24
JN 1350 DATA 173,255,6,125,248,6,141,255,
   6,24,185,16,6,125,216,6,153,16,6,221,2
   08,6,208,6
YF 1360 DATA 189,200,6,153,16,6,200,200,2
   00,232,224,7,144,201,173,255,6,141,0,2
   08,32,57,6,32,88,6,76,98,228
ZT 1370 REM DISPLAY LIST IN PAGE 6
PG 1380 RESTORE 1390:FOR J=1536 TO 1679:R
   EAD A:POKE J,A:NEXT J
LW 1390 DATA 112,112,112,70,0,0,68,20,0,6
   8,60,0,69,100,0,69,0,0,69,80,0,69,128,
   0,69,208,0,69,0,0,69,80,0
AN 1400 DATA 69,128,0,69,140,0,68,180,0
DL 1410 DATA 68,0,0,70,40,0,65,0,6
AQ 1420 DATA 0,0,0,0,0,0,172,55,6,240,23,
   206,56,6,16,18,177,208,141,6,210,200
HH 1430 DATA 177,208,240,5,141,56,6,200,1
   52,141,55,6,96
UE 1440 DATA 0,0,172,86,6,240,40,206,87,6
   ,48,17,169,13,205,87,6,144,3,173,87,6,
   9,160,141,5,210,208,18,177,208
AI 1450 DATA 141,4,210,200,177,208,240,5,
   141,87,6,200,152,141,86,6,96,104,160,0
   ,162,0,169,7,76,92,228
CH 1460 RESTORE 1470:FOR J=1541 TO 1583 S
   TEP 3:READ A:POKE J,HI+A:NEXT J
VX 1470 DATA 2,2,2,2,0,0,0,0,1,1,1,2,2,3,
   3
KV 1480 RESTORE 1490:FOR J=1736 TO 1791:R
   EAD A:POKE J,A:NEXT J
FL 1490 DATA 0,80,128,208,0,80,128,0,16,6
   4,144,192,16,64,144,0,1,255,1,255,1,25
```

```
   5,1,0
LW 1500 DATA 20,12,9,12,15,18,21,0,0,0,0,
   0,0,0,0,0,20,12,9,12,15,18,21,0,252,4,
   252,4,252,4,252,120
ES 1510 HV=INT(VBI/256):POKE 1674,HV:POKE
    1672,VBI-256*HV
AP 1520 RETURN
WE 1525 REM PM IMAGES
RP 1530 RESTORE 1540:FOR J=1 TO 12:READ A
   :CR$(J,J)=CHR$(A):NEXT J
XZ 1540 DATA 231,60,90,60,24,60,90,153,24
   ,36,36,102
EX 1550 RESTORE 1560:FOR J=1 TO 12:READ A
   :CRJ$(J,J)=CHR$(A):NEXT J
YF 1560 DATA 66,36,60,90,60,153,126,24,24
   ,60,66,195
FQ 1570 BL$(1)=CHR$(0):BL$(13)=CHR$(0):BL
   $(2)=BL$
ZZ 1580 RESTORE 1590:FOR J=1 TO 12:READ A
   :CRF1$(J,J)=CHR$(A):NEXT J
KU 1590 DATA 195,36,60,90,60,25,62,88,156
   ,36,38,96
WL 1600 RESTORE 1610:FOR J=1 TO 12:READ A
   :CRF2$(J,J)=CHR$(A):NEXT J
EO 1610 DATA 195,36,60,90,60,152,124,26,5
   7,36,100,6
IF 1620 RESTORE 1630:FOR J=1 TO 12:READ A
   :CRM$(J,J)=CHR$(A):NEXT J
JW 1630 DATA 66,165,60,90,36,24,126,153,6
   0,126,36,102
BH 1640 RESTORE 1650:FOR J=1 TO 12:READ A
   :HT1$(J,J)=CHR$(A):NEXT J
LS 1650 DATA 0,216,248,248,112,32,0,27,31
   ,31,14,4
EJ 1660 RESTORE 1670:FOR J=1 TO 12:READ A
   :HT2$(J,J)=CHR$(A):NEXT J
XW 1670 DATA 0,27,31,31,14,4,0,216,248,24
   8,112,32
ZK 1680 RESTORE 1690:FOR J=1 TO 12:READ A
   :CUR$(J,J)=CHR$(A):NEXT J
XP 1690 DATA 68,238,254,254,254,254,254,1
   24,124,56,56,16
NZ 1700 REM PM INIT.
ZW 1710 P0$(1)=CHR$(0):P0$(256)=CHR$(0):P
   0$(2)=P0$:YPOS=187:P0$(YPOS)=CR$
BB 1720 P1$(1)=CHR$(0):P1$(256)=CHR$(0):P
   1$(2)=P1$:P2$(1)=CHR$(0):P2$(256)=CHR$
   (0):P2$(2)=P2$
CV 1730 P3$(1)=CHR$(0):P3$(256)=CHR$(0):P
   3$(2)=P3$:P1$(60)=CRM$:P2$(44)=CR$:JPO
   S=100:DELJ=3:RTLJ=200:LLJ=50:DELTA=6
OH 1740 POKE 54279,HI:POKE 559,62:POKE 53
   277,3:POKE 53248,120:POKE 704,116:POKE
    53249,126:POKE 705,92
VW 1750 POKE 53250,100:POKE 706,20:POKE 7
   07,0:POKE 623,1
BF 1760 RETURN
EF 1770 REM PACKAGES ON BELTS
OK 1780 DD$(1)="e":DD$(448)="e":DD$(2)=DD
   $
PY 1790 FOR J=1760 TO 1766:SPEED=RND(0)*(
   9-DIF)*4+(7-DIF):IF DIF>3 THEN SPEED=S
   PEED*1.8
XM 1792 POKE J,SPEED:NEXT J
HS 1800 FOR L=0 TO 6:RESTORE 1850+10*L:RE
   AD F$:FOR K=0 TO 3
VO 1810 POS=64*L+16*K:DD$(POS+1)=F$
YS 1820 IF DIF>3 THEN DD$(POS+9)=F$
PP 1840 NEXT K:NEXT L:RETURN
DD 1850 DATA ACB
YM 1860 DATA ACB
GC 1870 DATA DDD
EH 1880 DATA EFG
JE 1890 DATA EFG
AP 1900 DATA DDD
II 1910 DATA EFG
```

how the pros enhance Atari music

16-BIT SOUNDPOWER

Article on page 38

LISTING 1

Don't type the TYPO II Codes!

```
OT 10 REM SIXTEEN-BIT SOUND DEMO
AU 20 REM BY JERRY WHITE
FW 30 REM (c) 1985, ANTIC PUBLISHING
CA 100 RESTORE
AH 110 GRAPHICS 2+16:? #6:? #6;"     16-BI
   T SOUND":? #6:? #6;"   BY JERRY WHITE"

WB 120 DIM S16$(56):REM HOLDS ML SUBROUTI
   NE
QM 130 FOR ME=1 TO 56:READ BYTE:S16$(ME,M
   E)=CHR$(BYTE):NEXT ME
QD 150 SOUND 0,0,0,0:POKE 53768,120
NH 160 DIM N$(24),FREQ(7,12):N$="B A#A G#
   G F#F E D#D C#C ":GOTO 220
EK 170 REM DISPLAY SUBROUTINE
SP 180 IF BOTH THEN POSITION 5,5:? #6;"OC
   TAVES: ";OCTAVE;"&";OCTAVE+1;:GOTO 190

GT 185 POSITION 5,5:? #6;"OCTAVE: ";OCTAV
   E;
JP 190 POSITION 5,7:? #6;"PITCH: ";PITCH;
   " ";
BI 200 SP=PITCH*2-1:POSITION 5,9:? #6;"NO
   TE: ";N$(SP,SP+1):RETURN
SP 210 REM CREATE FREQUENCY ARRAY
MV 220 FOR OCTAVE=7 TO 1 STEP -1:FOR PITC
   H=12 TO 1 STEP -1
NI 230 READ FREQ:FREQ(OCTAVE,PITCH)=FREQ:
   NEXT PITCH:NEXT OCTAVE
JH 240 BOTH=0:LOWOCT=7:WAIT=10
HV 250 REM MAIN SOUND LOOPS
OI 260 FOR OCTAVE=LOWOCT TO 1 STEP -1:FOR
    PITCH=12 TO 1 STEP -1
HA 270 SETCOLOR 4,PITCH,0:GOSUB 180:VOL=8
   :POKE 540,VOL
DW 280 IF  NOT BOTH THEN JW=USR(ADR(S16$)
   ,FREQ(OCTAVE,PITCH),VOL):GOTO 300
QB 290 JW=USR(ADR(S16$),FREQ(OCTAVE,PITCH
   ),VOL,FREQ(OCTAVE+1,PITCH),VOL)
SH 300 IF  NOT VOL THEN 320
AY 310 VOL=PEEK(540):GOTO 280
VR 320 GOSUB 480
YP 330 NEXT PITCH:NEXT OCTAVE
QW 340 IF  NOT BOTH THEN BOTH=1:LOWOCT=6:
   GOTO 260
DY 350 FOR ME=5 TO 9 STEP 2:POSITION 5,ME
   :? #6;"                 ";:NEXT ME
FM 370 VOL=8:OCTAVE=5:PITCH=7:HOLD=16:WAI
   T=8:GOSUB 530
```

continued on next page

```
YB 380 PITCH=12:HOLD=4:WAIT=0:GOSUB 530
DB 390 OCTAVE=6:PITCH=1:HOLD=4:WAIT=0:GOS
   UB 530
YQ 400 OCTAVE=5:PITCH=12:HOLD=4:WAIT=0:GO
   SUB 530
IR 410 PITCH=10:HOLD=16:WAIT=8:GOSUB 530
EK 420 PITCH=12:HOLD=16:WAIT=32:GOSUB 530

FB 430 PITCH=8:HOLD=16:WAIT=8:GOSUB 530
VE 440 PITCH=7:HOLD=32:WAIT=0:GOSUB 530
KT 450 GRAPHICS 0:END
BU 460 REM TIME DELAY
JQ 480 POKE 540,WAIT
FY 490 IF PEEK(540) THEN 490
ZB 500 RETURN
MG 510 REM SUBROUTINE TO CONTROL
RY 520 REM TWO 16-BIT SOUNDS
VC 530 POKE 540,HOLD:X=USR(ADR(S16$),FREQ
   (OCTAVE,PITCH),VOL,FREQ(OCTAVE+1,PITCH
   ),VOL)
BW 540 IF PEEK(540) THEN 540
MV 550 X=USR(ADR(S16$),FREQ(OCTAVE,PITCH)
   ,0,FREQ(OCTAVE+1,PITCH),0)
QK 560 GOTO 480
JR 20000 REM DATA FOR M-L SUBROUTINE
WV 20010 DATA 104,201,2,240,33,201,4,240,
   12,170,224,0,240,41
TD 20020 DATA 202,104,104,240,247,208,245
   ,104,141,2,210,104,141,0
GX 20030 DATA 210,104,104,41,15,9,160,141
   ,3,210,104,141,6,210
IG 20040 DATA 104,141,4,210,104,104,41,15
   ,9,160,141,7,210,96
BU 30000 REM FREQUENCIES FOR FREQ ARRAY
NT 30010 DATA 27357,25821,24372,23003,217
   12,20493,19342,18256,17231,16264,15351
   ,14489
HW 30020 DATA 13675,12907,12182,11498,108
   52,10243,9668,9125,8612,8128,7626,7241

EP 30030 DATA 6834,6450,6088,5746,5423,51
   18,4830,4559,4303,4061,3832,3617
WT 30040 DATA 3414,3222,3040,2869,2708,25
   55,2412,2276,2148,2027,1913,1805
JN 30050 DATA 1703,1607,1517,1431,1350,12
   74,1202,1134,1070,1010,953,899
FK 30060 DATA 848,800,755,712,672,634,598
   ,564,532,501,473,446
IK 30070 DATA 421,397,374,353,332,313,295
   ,278,262,247,233,219
```

92 chess solutions in 40 seconds

8 QUEENS ACTION!

Article on page 40

LISTING 1

```
; 8-QUEENS SOLUTION
; BY DAVE OBLAD
; (c) 1985, ANTIC PUBLISHING

BYTE ARRAY T(96),P(8),I(8),O(8),M(8)
BYTE A,B,C,D,X,Y,L1,L2,L3,OPT=53279

PROC SEARCH()
 FOR X=0 TO D
  DO
   Y=X*8 B=1
   FOR A=0 TO 7
    DO
     IF T(Y+A)#O(A) THEN B=0 FI
    OD
   IF B=1 THEN RETURN FI
  OD
RETURN

PROC ROTATE()
 FOR A=0 TO 7
  DO
   B=7-O(A) M(B)=A
  OD
 FOR A=0 TO 7
  DO
   O(A)=M(A)
  OD
RETURN

PROC TEST()
 FOR A=0 TO 7
  DO O(A)=P(A) OD
 FOR L1=0 TO 1
  DO
   FOR L2=0 TO 1
    DO
     FOR L3=0 TO 3
      DO
       SEARCH()
        IF B=1 THEN RETURN FI
       ROTATE()
      OD
     FOR A=0 TO 7
      DO M(A)=O(A) OD
     FOR A=0 TO 7
      DO O(7-A)=M(A) OD
    OD
   FOR A=0 TO 7
    DO O(A)=7-O(A) OD
  OD
 B=0
RETURN

PROC KEEP()
 X=D*8
 FOR A=0 TO 7
  DO T(X+A)=P(A) OD
RETURN

PROC DISPLAY()
;REMOVE 5 SEMI-COLONS BELOW
;FOR UNIQUE SOLUTIONS ONLY!

;IF D#0 THEN TEST()
; IF B=1 THEN RETURN
;  ELSE KEEP()
; FI
;FI

 FOR Y=0 TO 7
  DO
   FOR X=0 TO 7
    DO
     POSITION(X+15,Y+8)
     IF P(Y)=X THEN PRINT("Q")
```

```
    ELSE PRINT("+") FI
   OD
  OD
 POSITION(18,18)
 D==+1 PRINTB(D)
RETURN

PROC TRY()
 FOR Y=0 TO 6
  DO
   FOR X=Y+1 TO 7
    DO
     A=P(X)-P(Y) B=X-Y
     IF A>7 THEN A=255-A+1 FI
     IF A=B THEN RETURN FI
    OD
  OD
 DISPLAY()
RETURN

PROC SWAP()
 C=0 I(C)==+1
 WHILE I(C)=C+2
  DO
   I(C)=0 C==+1 I(C)==+1
   IF C<7 THEN
    FOR B=0 TO C
     DO
      A=P(B) P(B)=P(B+1) P(B+1)=A
     OD
    FI
  OD
 A=P(0) P(0)=P(1) P(1)=A
RETURN

PROC MAIN()
BYTE CONSOLE=53279

DO
 GRAPHICS(0) POKE(752,1)
 POSITION(8,0)
 PRINTE("  8-QUEENS SOLUTIONS")
 PRINTE("           BY DAVE OBLAD")
 FOR A=0 TO 7 DO P(A)=A I(A)=0 OD
 FOR A=0 TO 96 DO T(A)=0 OD
 D=0
 DO
  TRY() SWAP()
   FOR A=0 TO 7
    DO
     IF A#P(A) THEN EXIT FI
    OD
   IF A=8 OR OPT#7 THEN EXIT FI
 OD
  IF A=8 THEN POSITION(15,20)
   PRINTE("COMPLETE")
   PUTE()
   PRINTE("PRESS START TO RE-RUN")
  FI
 DO
  UNTIL CONSOLE < 7
 OD
OD
RETURN
```

assembly language

FINE SCROLLING WORLD: PART I

Article on page 70

LISTING 1

```
0 ; COARSE SCROLLING, LISTING 1
10 ; BY MARK ANDREWS
20 ; ANTIC PUBLISHING
30 ;
40        *=  $3000
45        JMP INIT
50 ;
60 SDMCTL =  $022F
70 ;
80 SDLSTL =  $0230
90 SDLSTH =  $0231
0100 ;
0110 COLOR0 = $02C4   ;OS COLOR REGISTE
R
0120 COLOR1 = $02C5
0130 COLOR2 = $02C6
0140 COLOR3 = $02C7
0150 COLOR4 = $02C8
0160 ;
0170 TCKPTR = $2000
0180 ;
0190      .OPT OBJ
0200 ;
0210 ; DISPLAY LIST DATA
0220 ;
0230 START
0240 LINE1 .SBYTE "   ANTIC PRESENTS
 "
0250 LINE2 .SBYTE "
 "
0260      .SBYTE "    coarse scrolling
 "
0270 LINE3 .SBYTE "                 On You
r"
0280      .SBYTE " Atari                 "
0290 LINE4 .SBYTE "   BY (YOUR NAME)
 "
0300 ;
0310 ; DISPLAY LIST
0320 ;
0330 HLST NOP          ;('HELLO' LIST)
0340      .BYTE $70,$70,$70
0350      .BYTE $70,$70,$70,$70,$70
0360      .BYTE $46
0370      .WORD LINE1
0380      .BYTE $70,$70,$70,$70,$47
0390 SCROLN NOP        ;(THIS IS THE LIN
E WE'LL SCROLL)
0400      .WORD $00    ; A BLANK TO BE F
ILLED IN LATER
0410      .BYTE $70,$42
0420      .WORD LINE3
0430      .BYTE $70,$70,$70,$70,$46
0440      .WORD LINE4
```

continued on next page

```
0450      .BYTE $70,$70,$70,$70,$70
0460      .BYTE $41
0470      .WORD HLST
0480 ;
0490 ; RUN PROGRAM
0500 ;
0510 INIT NOP          ;PREPARE TO RUN P
ROGRAM
0520      LDA COLOR3   ; SET COLOR REGIS
TER
0530      STA COLOR1
0540      LDA COLOR4
0550      STA COLOR2
0560 ;
0570      LDA #0       ; TELL ANTIC WHER
E DISPLAY LIST IS
0580      STA SDMCTL
0590      LDA #HLST&255
0600      STA SDLSTL
0610      LDA #HLST/256
0620      STA SDLSTH
0630      LDA #$22
0640      STA SDMCTL
0650 ;
0660 ; COARSE-SCROLLING ROUTINE
0670 ;
0680      LDA #42      ; # OF CHARACTERS
 IN SCROLL LINE
0690      STA TCKPTR
0700      JSR TCKSET
0710 ;
0720 COARSE
0730      LDY TCKPTR   ; NUMBER OF CHARA
CTERS IN SCROLL LINE
0740      DEY
0750      BNE SCORSE   ; LOOP BACK UNTIL
 FULL LINE IS SCROLLED
0760      LDY #42      ; NUMBER OF CHARA
CTERS TO SCROLL
0770      JSR TCKSET
0780 SCORSE NOP        ;DO COARSE SCROLL
0790      STY TCKPTR
0800      INC SCROLN   ; LOW BYTE OF ADD
RESS
0810      BNE LEAP
0820      INC SCROLN+1 ; HIGH BYTE OF A
DDRESS
0830 ;
0840 ; DELAY  LOOP
0850 ;
0860 LEAP
0870      TYA
0880      PHA          ; SAVE Y REGISTER
0890      LDX #$FF
0900 XLOOP
0910      LDY #$80
0920 YLOOP
0925      DEY
0930      BNE YLOOP
0940 ;
0950      DEX
0960      BNE XLOOP
0970      PLA
0980      TAY          ; RESTORE Y REG
0990 ;
1000      JMP COARSE
1010 ;
1020 TCKSET
1030      LDA #LINE2&255
1040      STA SCROLN
1050      LDA #LINE2/256
1060      STA SCROLN+1
1070      RTS
1080      *=  $02E0
1090      .WORD INIT
```

LISTING 2

```
0 ; COARSE SCROLLING, LISTING 2
10 ; BY MARK ANDREWS
20 ; ANTIC PUBLISHING
0190 ;
0240 LINE1 .BYTE $00,$00,$00,$21,$2E,$
34,$29,$23,$00
0245       .BYTE $30,$32,$25,$33,$25,$2E
,$34,$33,$00,$00,$00
0250 LINE2 .BYTE $00,$00,$00,$00,$00,$
00,$00,$00,$00,$00,$00
0255       .BYTE $00,$00,$00,$00,$00,$00
,$00,$00,$00,$00,$00,$00
0260       .BYTE $00,$00,$00,$00,$63,$6F
,$61,$72,$73,$65,$00,$73,$63,$72
0265       .BYTE $6F,$6C,$6C,$69,$6E,$67
,$00,$00,$00,$00,$00,$00,$00,$00
0270 LINE3 .BYTE $00,$00,$00,$00,$00,$
00,$00,$00,$00,$00,$00,$00,$00
0275       .BYTE $2F,$6E,$00,$39,$6F,$75
,$72
0280       .BYTE $00,$21,$74,$61,$72,$69
,$00,$00,$00,$00,$00
0285       .BYTE $00,$00,$00,$00,$00,$00
,$00,$00,$00
0290 LINE4 .BYTE $00,$00,$00,$22,$39,$
00,$08,$39,$2F,$35,$32
0295       .BYTE $00,$2E,$21,$2D,$25,$09
,$00,$00,$00
```

---

starting out

SOUND EFFECTS LIBRARY

Article on page 11

LISTING 1

Don't type the TYPO II Codes!

```
UV 10 REM SPECTACULAR SOUND
MX 20 REM BY TIMOTHY BANSE
FW 30 REM (c) 1985, ANTIC PUBLISHING
RU 40 DIM A$(24)
VV 50 GRAPHICS 0:RESTORE 100:? ,"AVAILABL
   E SOUNDS:":?
PE 60 FOR X=1 TO 17
KX 70 READ A$
YM 80 ? "(";X;") ";A$
OC 90 NEXT X
GG 100 DATA FACTORY WHISTLE,STEAM LOCOMOT
   IVE,BUZZER,FALLING OBJECT,SPACE SHIP,A
   RTIFICIAL INTELLIGENCE
KY 110 DATA OLD AIRPLANE,SINGING BIRD,LAS
   ER WEAPON,OCEAN SURF,BUSY SIGNAL,EXPLO
   SION,ALARM
XP 120 DATA GALAXY CANTINA MUSIC,CRASHING
    PLANE,PLAY A CHORD,THUNDER
```

```
JH 130 TRAP 50:? :? "Your Selection";:INP
   UT S
UZ 140 POKE 755,1:SETCOLOR 2,(S-1),2
AL 150 ON S GOSUB 180,260,350,400,450,500
   ,540,590,650,730,840,940,1000,1070,114
   0,1240,1430
RC 160 GOTO 50
NA 170 REM FACTORY WHISTLE
VJ 180 SOUND 0,60,10,8
WJ 190 SOUND 1,70,10,8
KB 200 SOUND 2,10,8,2
GW 210 FOR I=1 TO 1000
FW 220 NEXT I
DN 230 SOUND 0,0,0,0:SOUND 1,0,0,0
SC 240 SOUND 2,0,0,0:RETURN
AC 250 REM STEAM LOCOMOTIVE
YE 260 FOR I=1 TO 100
SR 270 SOUND 0,80,2,11
HI 280 FOR DELAY=1 TO 10
TE 290 NEXT DELAY
VN 300 SOUND 0,0,0,0
GV 310 FOR DELAY=1 TO 10
SR 320 NEXT DELAY
MP 330 NEXT I:RETURN
GN 340 REM BUZZER
SH 350 SOUND 0,40,6,10
YD 360 FOR DELAY=1 TO 400
TB 370 NEXT DELAY
RP 380 SOUND 0,0,0,0:RETURN
OK 390 REM FALLING OBJECT
OY 400 FOR X=30 TO 200
CV 410 SOUND 0,X,10,8
LS 420 NEXT X
RG 430 SOUND 0,0,0,0:RETURN
BS 440 REM SPACE SHIP
WA 450 FOR X=20 TO 255
WK 460 SOUND 0,X,2,8
UU 470 FOR DELAY=1 TO 3:NEXT DELAY
UB 480 NEXT X:SOUND 0,0,0,0:RETURN
OX 490 REM ARTIFICIAL INTELLIGENCE
DA 500 FOR X=1 TO 100
ZK 510 SOUND 0,INT(RND(1)*75),12,8
TQ 520 NEXT X:SOUND 0,0,0,0:RETURN
FY 530 REM OLD AIRPLANE
DY 540 FOR X=1 TO 200
BQ 550 SOUND 0,99,10,8
WB 560 SOUND 0,0,0,0
ST 570 NEXT X:RETURN
JM 580 REM BIRD SINGING
RH 590 FOR I=1 TO 10
XB 600 FOR X=1 TO 30
FJ 610 SOUND 0,X,14,8
LU 620 NEXT X
NZ 630 NEXT I:SOUND 0,0,0,0:RETURN
RP 640 REM LASER WEAPON
OC 650 FOR I=1 TO 5
DG 660 FOR X=1 TO 75
WO 670 SOUND 0,X,2,8
MG 680 NEXT X
WI 690 SOUND 0,0,0,0
MN 700 FOR DELAY=1 TO 50:NEXT DELAY
MP 710 NEXT I:RETURN
HN 720 REM OCEAN SURF
YD 730 FOR X=0 TO 50
ZV 740 SOUND 0,X,8,8
SH 750 FOR I=1 TO 30
GJ 760 NEXT I
MF 770 NEXT X
VA 780 FOR X=10 TO 0 STEP -1
XP 790 SOUND 0,X,8,4
XY 800 FOR I=1 TO 100
GA 810 NEXT I
TT 820 NEXT X:SOUND 0,0,0,0:RETURN
FX 830 REM TELEPHONE BUSY SIGNAL
WF 840 FOR X=1 TO 10
WF 850 SOUND 0,50,12,8
WP 860 SOUND 1,0,0,0
CM 870 FOR DELAY=1 TO 100:NEXT DELAY
WI 880 SOUND 0,0,0,0
XC 890 SOUND 1,40,13,8
BZ 900 FOR DELAY=1 TO 100:NEXT DELAY
LV 910 NEXT X
RU 920 SOUND 1,0,0,0:RETURN
PL 930 REM EXPLOSION
OE 940 FOR TONE=15 TO 0 STEP -1
UR 950 SOUND 0,25,16,TONE
KT 960 FOR DELAY=1 TO 20:NEXT DELAY
KR 970 NEXT TONE
RV 980 SOUND 0,0,0,0:RETURN
OE 990 REM WARNING ALARM
YB 1000 FOR I=1 TO 5
DG 1010 FOR N=35 TO 97
WH 1020 SOUND 0,N,10,15
HB 1030 NEXT N
TH 1040 FOR DELAY=1 TO 5
MU 1050 NEXT I:SOUND 0,0,0,0:RETURN
PL 1060 REM GALAXY CANTINA MUSIC
XC 1070 FOR I=0 TO 3
ZF 1080 FOR X=15 TO 0 STEP -0.2
UD 1090 SOUND 0,X,12,8
VL 1100 FOR DELAY=1 TO 5:NEXT DELAY
LD 1110 NEXT X
MN 1120 NEXT I:SOUND 0,0,0,0:RETURN
AO 1130 REM PLANE CRASHING
OC 1140 FOR X=255 TO 40 STEP -1
AW 1150 SOUND 0,X,14,10
EC 1160 FOR I=1 TO 20
GS 1170 NEXT I:NEXT X
CR 1180 FOR X=15 TO 0 STEP -1
PB 1190 SOUND 0,25,16,X
DM 1200 FOR I=1 TO 20
GC 1210 NEXT I:NEXT X
HK 1220 RESTORE 1260:RETURN
GR 1230 REM PLAY A CHORD
LZ 1240 REM GET NOTES TO PLAY
EK 1250 READ ONE,TWO,THREE,FOUR
CA 1260 DATA 100,200,50,150
YU 1270 RESTORE 1260:FOR TIMES=1 TO 3
GW 1280 REM EACH NOTE GETS OWN VOICE
QD 1290 SOUND 0,ONE,10,8
DR 1300 SOUND 1,TWO,10,8
EW 1310 SOUND 2,THREE,10,8
HG 1320 SOUND 3,FOUR,10,8
OT 1330 REM HOLD NOTES FOR A BEAT
FP 1340 FOR DELAY=1 TO 100:NEXT DELAY
EG 1350 REM TURN OFF EACH VOICE/NOTE
HU 1360 SOUND 0,0,0,0
IJ 1370 SOUND 1,0,0,0
IY 1380 SOUND 2,0,0,0
JN 1390 SOUND 3,0,0,0
NU 1400 REM REPEAT THE CHORD 3 TIMES
QZ 1410 NEXT TIMES:RETURN
BJ 1420 REM THUNDER
AS 1430 FOR RUMBLE=1 TO 3
AJ 1440 Y=INT(255*RND(1)+20)
DP 1450 X=RND(1)*150
YT 1460 FOR PITCH=1 TO Y
IW 1470 SOUND 0,PITCH,8,15
GC 1480 NEXT PITCH
YZ 1490 FOR DELAY=1 TO X:NEXT DELAY
GQ 1500 NEXT RUMBLE:SOUND 0,0,0,0:RETURN
```

FINE SCROLLING WORLD: PART I

Speedy, smooth moves in assembly language

by MARK ANDREWS

*Learn how to fine scroll with all the speed and power of machine language. This is the start of a two-part tutorial by Mark Andrews, author of **Atari Roots**. The demo program is written in assembly language and requires the MAC/65 assembler or the Atari Assembler Editor. It will run on any 8-bit Atari computer, with disk or cassette. The article is intended for those with at least an introductory knowledge of assembly language.*

PART 1: COARSE SCROLLING

*Mark Andrews wrote what is probably the finest introduction to Atari assembly language: **Atari Roots**. So popular was this book that it sold out and is currently out of print. **Antic** published excerpts from the book in November and December 1984. We continue this practice with the following two-part tutorial which was specially adapted from the book by Mark for* **Antic**. —ANTIC ED

If you haven't been able to find a good fine scrolling program written in BASIC, I can tell you why: There's no such thing!

To implement fine scrolling on an Atari computer, you have to shift the position of every dot on the screen 60 times per second. BASIC is far too slow to handle that kind of data-juggling.

So when you try to write a fine scrolling program in BASIC, what you usually wind up with is a lot of jerking, jumping and smearing on your video screen. To take advantage of the fine scrolling capabilities of your Atari computer, you have to use—you guessed it—assembly language.

In this month's article, I'll explain display lists and coarse scrolling using a type-in assembly language listing. Next time we'll go into fine scrolling and the use of vertical blank interrupts.

DISPLAY LIST

To understand how coarse scrolling and fine scrolling work, it's necessary to have at least a fundamental understanding of a programming technique called display list modification.

In Atari computing, a display list is a special kind of data table that is used to set up a screen display. Display list modification is a method for altering a display list to suit the needs of an individual program.

Actually, a display list is a kind of program within a program. When a display list is included in an assembly language program, it can then be used as a program in its own right by ANTIC, the intelligent graphics chip built into every Atari computer. This programmable chip has just one job—generating video displays. To carry out this job, the ANTIC chip must always have access to a display list of some kind.

Several steps are involved in designing a customized display list. First you have to create the list, and store it somewhere in RAM. Then you have to POKE the address of the list, low byte first, into two memory registers in your computer—memory registers $0230 and $0231.

THE PROGRAM

Before we examine Listing 1, let's talk about typing it in. **MAC/65** owners should type in Listing 1 exactly as printed, then assemble and RUN it according to MAC/65 instructions. Those with **Atari Assembler Editor** should use the lines in Listing 2 instead of the corresponding lines in Listing 1.

Antic Disk subscribers: You will find the MAC/65 source code on disk as COARSE.M65. If you own the Assembler Editor, again, type in the lines from Listing 2. We have also included the assembled object code under the filename COARSE.EXE. To run the assembled object code from DOS 2, type [L] [RETURN] COARSE.EXE [RETURN].

DISPLAY LIST TAKE-APART

Examine Listing 1 and you'll see a customized display list in lines 330 through 470. As you'll discover when you RUN the program, this list is designed to display three different text modes on a screen simultaneously. Before the display list can be used, however, its address must be stored in memory registers $0230 and $0231. This is done in lines 570 through 640.

Now let's take a look at what happens when your computer's ANTIC chip encounters the display list. Look at the display list beginning at line 340. The first eight bytes it contains are identical, the value of each byte is the hex number $70.

When ANTIC encounters the number $70 in a display list, it prints one blank Graphics 0 line on the screen. Because of the overscan characteristics of television sets and video monitors, three of the blank lines in Listing 1 are out of viewing range at the top of the screen. So when you run the program in Listing 1, you'll see only three blank Graphics 0 lines at the top of your screen.

LOAD MEMORY SCAN

After these three blank lines are printed, the next value that your ANTIC chip will encounter is the number $46. In an Atari display list, a byte that begins with the digit 4 is always a code number called a Load Memory Scan (LMS) instruction. An LMS instruction tells the ANTIC chip two things.

First, it instructs ANTIC to display a line of text in a certain graphics mode. Then it tells ANTIC exactly where the text to be used on that line can be found. For example, the LMS instruction in line 360 orders ANTIC to display a line of GR. 1 text. (In ANTIC language, Mode 6 is the equivalent of BASIC's Graphics Mode 1). Then it tells ANTIC to look at the next two bytes of the display list for the starting address of the line of GR. 1 text which it is to display.

ADDRESS LABELS

In the display list we're looking at, the address of the text to be displayed is entered as the label LINE 1. In Listing 1, line 240 is labeled LINE 1. Look at line 240, and you'll see that it's the line which contains the words ". . . ANTIC PRESENTS . . ." (Atari Assembler Editor uses .BYTE hex internal characters instead.) So, when your computer's ANTIC chip encounters the LMS byte in line 360 of Listing 1, it will display the centered line "ANTIC PRESENTS" in GR. 1 on your video screen.

Examine the display list in Listing 1 further and you'll see several other LMS instructions. In line 410, there's an LMS byte that tells ANTIC to display the line "On Your Atari" in BASIC Text Mode 0 (ANTIC Mode 2). In line 430, there's an LMS instruction that instructs ANTIC to display the words "BY (YOUR NAME)" in BASIC Mode 1.

And in line 460, there's a special kind of LMS instruction which is used at the end of every display list. It orders ANTIC to loop back to the beginning of the display list every 1/60th of a second, and to display the list again.

Although you can't detect it by what I've told you so far, there's also a special kind of LMS instruction in line 380 of Listing 1. If you're a sharp-eyed reader, you may have noticed that the two bytes after this instruction are blank. You'll learn why later. . .

COARSE SCROLLING

Now that you know what a display list looks like, what a display list does, and a little bit about how it does it, we're ready to take a closer look at our program.

The most common way to implement coarse scrolling is to set up a loop that keeps incrementing or decrementing certain bytes in a display list—specifically the bytes that follow the LMS instructions for the lines that are to be scrolled.

As I've pointed out, the two bytes that follow an LMS instruction always contain the starting address of the text lines which the LMS instruction will display. So, to set up a coarse horizontal scroll, all you have to do is write a loop that progressively increments or decrements the byte that follows the LMS instruction for each line you want scrolled.

If your loop increments the bytes that follow your LMS instructions, your display will scroll from right to left. If your loop decrements those bytes, then your lines controlled by the LMS instruction will scroll from left to right.

If your scrolling program is written in assembly language, it will ordinarily have to include some sort of delay loop since machine language is so fast that it will cause a scroll line to zoom by in a blur.

continued on next page

In the coarse-scrolling program which I've written for this article, only one line of text is scrolled. But you can use coarse scrolling to scroll any number of lines you like, up to—and including—the maximum number of lines in a full-screen display. If you want to scroll your entire screen, you can precede every line with an LMS instruction!

VERTICAL SCROLLING

It's almost as easy to do coarse vertical scrolling as it is to do coarse horizontal scrolling. To scroll a screen display vertically, all you have to do is increment or decrement the LMS address of each line you want scrolled, by the number of characters in the lines being scrolled instead of by just one character at a time.

When you set up this kind of scrolling action, you have to count the number of characters in each line very carefully, so your characters won't move back and forth on the screen as they scroll up or down.

Next time we'll go on to actual fine scrolling, now that the groundwork has been laid.

Listing on page 67.

product reviews

STAR SR-10

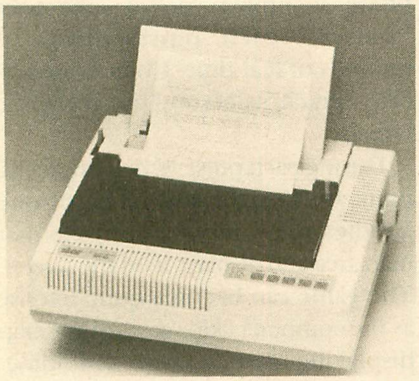
Star Micronics
200 Park Avenue
New York, NY 10166
(212) 986-6770
$649

Reviewed by Charles Jackson

July's **Antic** favorably reviewed the Star SG-10, successor to the Gemini 10X. Now comes the mid-line **SR-10** which is almost twice as fast, quieter, packed with features and compatible with most SG-10 software.

Other SR-10 features include a good-looking "near letter quality" mode, built-in 2K buffer (expandable to 6K), reverse paper feed for superscript, ten different character sets and room for 240 user-defined characters.

The rugged tractor feed assembly is hidden beneath a rear cover just behind the platen. It is invisible during normal operation.



Tractor assembly is easily bypassed for single-sheet operation, you won't have to wrestle with a stubborn "removable" tractor!

The SR-10 printhead uses a 9-wire, 9×11 dot configuration like the SG-10, and both are able to print quadruple density bit image graphics.

The SR-10 is also a paper miser. You only have to advance the paper one inch from the printhead to the paper tear bar. That's only one-fifth of the space required by the Gemini 10X and SG-10.

Our one complaint about the SR-10 is that it does not use standard typewriter ribbons like the two models above. This printer requires a special $9.95 ribbon cartridge similar to those used by Epson printers.

The 244-page manual accompanying the SR-10, though poorly organized and not indexed, is both comprehensive and instructive.

TRIVIA QUEST

Royal Software
2160 W. 11th Avenue
Eugene, OR 97402
(503) 683-5361
$39.95, 48K, 2 disks

Reviewed by Harvey Bernstein

Trivia Quest has much in common with other games of the genre. Playing against an opponent, you take turns answering questions of various values and difficulty in order to gain points. Players can determine the difficulty level of their questions, making for a fairer game. However, instead of being presented in the form of a dull Graphics 0 text screen, the game incorporates the questions into a race around a game board, a la Trivial Pursuit.

Players scramble to guide three pieces—Page, Knight, and Prince—around the board from their home castle. Each turn starts with the timed Wisdom Round. The faster you answer it, the more food you earn. The more food you have, the farther you can travel across the board. The first to bring all three men around, and earn the most gold points wins.

However, there are several complications. Each square on the board can represent a different category—TV & Movies, Sports & Entertainment, History & Geography, and Miscellaneous. The square you land on determines the category of your next question. In addition, there are flashing Dragon Squares. Landing on one of these brings you into an arcade mode, where you must shoot a dragon to gain additional gold.

The graphics and animation of the board and players are first-rate, making this one of the finer boardgame-to-computer translations I've seen. Unfortunately, this brings up the one feature of the game that I didn't like. You should be able to decide which of your three pieces to move each turn, based on the square it will land on. But that is impossible since the board *scrolls* and you can only see a small part of it at a time. On several occasions, I found that the category I landed on had no bearing on the question that came up. Also, in the games I played I've yet to see somebody succeed in killing the Dragon.

Trivia Quest comes on two double-sided disks—three sides containing question data. In addition, Royal Software sells a $24.95 Utility Disk that contains 1,000 additional questions plus a program to create your own question banks. Trivia Quest is the first trivia program to really take advantage of the special capabilities of the Atari. That by itself is enough to make it unique.

BANK STREET MUSIC WRITER

Mindscape, Inc.
3444 Dundee Road
Northbrook, IL 60062
(800) 221-9884
Illinois: (800) 942-7315
$49.95, 48K disk

Reviewed by Michael Lasky

Bank Street Music Writer will not only teach you the basic concepts of music, it also enables you to write and "play" music as well.

Of course it helps if you can already read music, but that isn't a pre-

continued on next page

requisite. Included in the clear and concise 64-page documentation is an introduction to music fundamentals that will teach the most tone-deaf neophyte such basics as notes, tempo, pitch and melody.

To demonstrate finished results, Music Writer has a dozen complete pieces of music on the flip side of the disk. These range from a long excerpt from Tchaikovski's "Nutcracker Suite," to Scott Joplin rags and "On Top of Old Smokey." This last song is used as the basis for the program's tutorial, which quickly teaches how to operate the keyboard commands and use them to start writing music.

The menu-driven program exploits the Atari's four-voice POKEY chip to its fullest capability. Each separate voice can be further enhanced and customized through a secondary [OPTION] menu. The various choices on the Main Menu screen are selected by using the cursor keys. To enter music, the New Piece option is selected. The work you do is saved on a separately formatted disk created with the special MusicDOS included with Music Writer.

To write notes on the screen music "paper," you position the cursor at the line corresponding to the note you want, and press a number key. Four is a quarter note, eight is an eighth note, and so on. If you make a mistake, the program's error protection alerts you and prevents the note from being entered. Notes already entered can be erased with a touch of the [SPACE] bar.

The music you have already notated can be heard at any time simply by pushing [START]. Each of the four separate voices is highlighted in different colors as each note is played, like a follow-along bouncing ball.

The only real weakness in this elaborate but easily-learned program is that the author has unwisely chosen a hard-to-read crimson red for the often-used edit screen. Other than this—which can be corrected by adjusting your TV set or monitor—the program is really worth getting keyed up about!

KAMPFGRUPPE

Strategic Simulations, Inc.
883 Stierlin Road, Bldg A-200
Mountain View, CA 94043
(415) 964-1353
$59.95, 48K disk

Reviewed by Dr. John Stanoch

Any wargamer will tell you that one of the most popular game subjects is the WWII Eastern Front. With the introduction of SSI's newest Atari game, **Kampfgruppe**, there are now eight computer wargames on the market dealing with this topic.

"All are grouped into 'combat formations'."

Kampfgruppe simulates platoon level combat from 1941 through 1945 on the Eastern Front. Each unit represents either a company or battalion of infantry, tanks, artillery, or transport vehicles. All are grouped into "combat formations."

One very strong feature of this game is that it allows players to construct their own scenarios using the whole gamut of weaponry employed by both sides during this conflict. Also, the players can create their own maps with any terrain placed on any desired square. However, there are also four widely varying historical scenarios provided for those who are not interested in building their own.

This game can be played by one, two, or even zero players. Although playing time is given as one to three hours, a new player can easily spend four or five hours playing while becoming familiar with the rules and game system. The game comes with a 23-page rulebook, a double-sided disk and a player reference data card.

The scrolling map is massive, containing over 12 screeens. While the background is plain black, the terrain features are rendered in high resolution color graphics. The tank silhouettes are particularly well done.

Movement on the map is across an "invisible" square grid, with units able to move horizontally, vertically and diagonally. Units are controlled by issuing keyboard input commands, assigning them each a movement or target objective. This may become tedious in larger scenarios in which each side may have more than 60 units apiece!

However, to aid the player, the game allows him to issue selected orders to an entire combat formation at a time, which may consist of 1 to 6 units. This feature speeds up the playing time considerably.

After all units are given orders, the combat phase executes them. This phase is broken down into four "pulses," each representing 30 seconds of real-time. Units automatically serach, select and fire at eligible targets.

During each pulse, when opposing units are in sight and range of one another, the computer jumps to that map area and displays it on the screen. The game can become quite intense as the computer skips around the map displaying key actions unfolding. Although the players have no direct control over the combat at this point, it is still very exciting to sit back and see how the battle develops.

My main criticism was that reading the rulebook was like studying a mathematics textbook abounding with formulae. These didactic explanations should have been separated from the rules proper and put into an appendix.

The game is an absolute must for any East Front devotee. But I would also recommend it to any serious wargamer. Kampfgruppe admirably

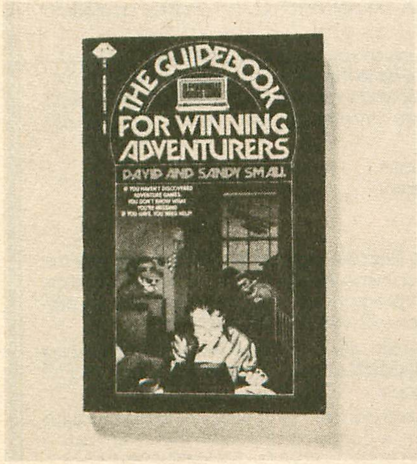
simulates the flux of mobile warfare and the importance of proper use of combined arms. Together with the excitement of the combat resolution, this makes the game well worth the hefty price.

Guidebook for Winning Adventurers

by David and Sandy Small
Baen Enterprises
8-10 W. 36th Street
New York, NY 10018
339 pages, paperbound
$9.95

Reviewed by Scott Lewis

Are you frustrated from repeatedly dying in the desert or disintegrating in outer space? Are you sick of mocking messages written by smart-aleck programmers with obviously twisted minds and the sick desire to humiliate others? Or are you merely running out of places to hide your hint books when friends come to call?



If any of the above descriptions fit you, now might be time to buy **The Guidebook for Winning Adventurers**. A good introduction to text adventures, this book provides specific tips and hints about six of the most popular Infocom games, Zork I, II and III, Enchanter, Infidel, and Planetfall.

The hints are all written in a simple code (b=a,c=b,a=z, etc.) so you won't accidentally see more clues than you need. The hints themselves are very good, progressing from a gentle nudge in the right direction to quite specific statements. Maps to the games are at the back of the book.

Lessons on winning methods of play are also offered. Mapping is covered in great detail and there is an interesting description of the programming techniques behind text adventures.

The authors are obviously dedicated and enthusiastic players of text adventures, and bring much of this enthusiasm to their book. Unfortunately, the text is not especially well-written and extracting the valuable information it contains can sometimes take a lot of effort.

In the end, though, the insights and help gained are well worth the exertion required. And ten bucks is little enough to pay, if it means not dying in that blasted desert again!

DR. WACKO'S ATARI BASIC

by David Heller and John Johnson
Addison-Wesley Publishing
Jacob Way
Reading, MA 01867
236 pages, paperbound
$12.95

Reviewed by Suzanne Clupper

Dr. Wacko Presents Atari BASIC and the Whiz-Bang Miracle Machine—to give the book's full name—is the latest in the well-known series of humorous but thorough introductions to BASIC programming for various computer brands.

The mythical Dr. Wacko introduces each Atari BASIC concept with plenty of examples to type in and try. Because so much is covered, no subject is treated with great depth. But there's more than enough to start out beginning programmers. Occasionally the humor becomes a bit much, but if you don't mind non-stop horrible puns, this is a good book to introduce you to the fundamentals of BASIC for the Atari 400/800, XL's and XE.

First we meet the keyboard, cursor movement, entering and editing text. Next come simple programming statements such as PRINT and REM, with examples to try in immediate mode.

"There's enough to start out beginning programmers."

This brings us to the real meat of BASIC, introducing variables, functions, strings, loops, subroutines and arrays—a tall order for one chapter. The chapter, called "The Great White Expanse," leads you on a quest through the desert from one oasis to another. At each oasis several new concepts are introduced.

There is so much information in this chapter that you even have to make sure you read all the cartoons in the margins, or else you might miss some necessary information. Another thing you might easily miss is the discussion of how to store and retrieve programs. It's in Appendix B, not the most likely place you'd expect to find such vital data.

A graphics chapter discusses screen pixels and then introduces graphics modes and the statements used to control them. The short chapter on sound explains the SOUND command and the concept of changing sounds by varying the pitch and volume.

The last chapter discusses programming style. The method given is to break down a problem into steps or

continued on page 77

ATARI®

$69.95 (Rom)
$49.95 (D or C)

Vastly SUPERIOR to any translation programs available! FOR ATARI 1200XL/600XL/800XL with 64K. (Please specify computer model number!)

ATARI®

$69.95 (Rom)
$49.95 (D or C)

THE XL "FIX"! ®

The Atari XL series computers represent power, sophistication, and flexibility virtually unrivalled in todays Home Computer Market.

With "approximately" 30-40% of existing software being "incompatable", a real, and serious problem exists. Because of this we have developed THE XL "FIX"!

ADVANTAGES over cheaper "translation products":

1. The XL "FIX"! is capable of fixing more software . . . an estimated **30% more software!**
2. The XL "FIX"! is available in **DISK, CASSETTE,** and now **ROM!**
3. XL "FIX"! versions fix ALL THREE types of software (Disk - Cassette - and Cartridges!)
4. The XL "FIX"! (disk or cassette) adds **OVER 4K** of usable RAM to your computer (anyone using Data bases or Word processors will **really** appreciate this feature!)
5. You never have to hold the OPTION button down on 600XL or 800XL computers!
6. VERY IMPORTANT! You need to load the XL "FIX"! only **once . . .** you can **change** disks, cassettes, or cartridges **without** rebooting the XL "FIX"! each time (disk or cassette)!
7. The **ROM** version is instantaneous upon computer power up, has a high speed cursor, is instantly switchable to your original operating system, will work with 16K 600XL's, and more!

The XL "FIX"! . . . . another SUPERIOR product! **64K required!**

DISTRIBUTOR/DEALER inquires welcome

Mastercard-Visa-Money Order or Cashier Check. Phone **(716) 467-9326** Please specify computer model number!

Send $49.95 ($69.95 for Rom) plus $4 shipping and handling (N.Y.S. residents please add 7%) to:
COMPUTER SOFTWARE SERVICES
P.O. Box 17660
Rochester, New York 14617

ATARI®

THREE NEW PRODUCTS!

ATARI®

THE "SUPER PILL"!

Exactly the same as the WORLD'S leading cartridge backup device... **THE PILL!**... except it's even simpler to operate, it's **SWITCHLESS!** Excellent for families having young children. Totally eliminates opening computer doors and switches. **THE "SUPER PILL"!** is the most advanced state of the **CARTRIDGE BACKUP** device available today. It is totally compatable with all ATARI computers and all programs backed up by the original "PILL"! Only $79.95 plus $4 shipping and handling.

THE "PROTECTOR/SILENCER"!

The **"PROTECTOR"!** is a disk and hardware modification (no soldering) for Atari 810, 1050, and Indus GT disk drives that will allow you to write true **BAD SECTORS** wherever you wish (not to be confused with ridiculous speed control or tape jerking schemes!). **Powerful** disk program finds hidden directories, scrambles existing directories, fast maps, hex conversions, disk dupes, and much more!

The **"SILENCER"!** quiets your drive tremendously (eliminates the **LOUD** grinding noise when you read a bad sector!), PLUS it allows you to **WRITE TO BOTH SIDES** of any disk **WITHOUT** cutting or notching the disk! Both for only $49.95 plus $4 shipping and handling.

THE "COMPANION"!

An amazing device that will enhance the capabilities of the **XL "FIX"!** or Atari Translater. It will allow you to **de-select BASIC** (no more need to hold the **OPTION** button while loading programs on the 600XL's and 800XL's), and it will allow you to **de-select the DIAGNOSTICS** (no more bad loads because of the DIAGNOSTICS jumping into the middle of your program load routine!). Installation is simple (10 minutes) and requires NO soldering! Only $29.95 plus $4 shipping and handling.

DISTRIBUTOR/DEALER inquiries welcome.

Our other fine products include THE "PILL"!, XL "FIX"!, "IMPOSSIBLE"!, "METAMORPHOSES"!, and "REMOTE"!

Mastercard-Visa-Money Order or Cashiers Check.
Phone orders:
(716) 467-9326.
Atari is a TM of Atari Inc. The "METAMORPHOSES"! is a TM of Computer Software Services (division of S.C.S.D., Inc.)

COMPUTER SOFTWARE SERVICES
P.O. Box 17660
Rochester, New York 14617

ATARI®

$149.95

For years they said it couldn't be done . . .

THE "IMPOSSIBLE"! ®

they claimed!

ATARI®

$149.95

Backup almost any disk currently available (even **heavily protected** programs) with an **UNMODIFIED** disk drive! Works with **ANY** disk drive!

PURPOSE: The **"IMPOSSIBLE"** was developed in response to the estimated half million disk drive users that own a drive other than the Atari 810 (Indus, Percom, Trak, Rana, Astra, etc.) that wish to **BACK UP** their **protected software.** Due to a radically new technology developed by Computer Software Services, modification to your disk drive has been eliminated! The advantages are obvious! Drive warranties are not violated, the chance accidental damage has been eliminated, etc., etc.

OPERATION: The **"IMPOSSIBLE"!** consists of a disk program (**unprotected** so you can make as many backups as you wish) and a 4K STATIC RAM pack which is inserted into your computer (no soldering!) The **"IMPOSSIBLE"!** will read your program disk and then **re-write it in an unprotected format!** You may make additional backup copies using a sector copier or even regular DOS! Because your backup copy no longer has BAD SECTORS or EXOTIC FORMATS, the program data can now be manipulated into DOS compatable files (even double density!), transfered to cassette, etc. (with the aid of our Satellite programs!) No user programming knowledge required. A few programs require logical thinking.

FEATURES:
1. Backup protected disks
2. Handles most MULTI-LOAD programs
3. Makes DOS files (with Satellite option)
4. Up to 90K data input capable
5. AFSD-Automatic FUZZY Sector Discriminator
6. Expands computer memory to 52K usable
7. Simple NO SOLDER installation
8. Satellite expandable

PROJECTED SATELLITES: A **"COMPACTOR"** program which will convert your program into DOS compatable files (double density compatable!) for the storage of several programs on one disk. A **"COLUMN 80"** program for Word Proccessing, etc. It allows 80 columns on the screen! The "XL-MATE" will allow programs made with your 400/800 "IMPOSSIBLE"! to now play on your XL Computer! The **METAMORPHOSES II** program will allow you to convert your **protected CASSETTES** into disk DOS files and vice-versa. All satellite programs must be used with inconjunction with The **"IMPOSSIBLE"!**

REQUIREMENTS: The **"IMPOSSIBLE"** diskette, the 4K STATIC RAM pack, a 400 or 800 computer (please specify!) with 48K and "B" Rom's. NOTE! The very old ATARI computers were shipped with "A" Rom's which had some serious "Bugs". Even if you don't own an **"IMPOSSIBLE,"** you should upgrade to "B" Rom's (simple to install!) We have them available at a very inexpensive price. CALL US! "XL" version available soon!

NOT A PIRATING TOOL: We at C.S.S. did not design The **"IMPOSSIBLE"!** to put Software Manufactures out-of-business overnight! Nearly all of **our** products have been "ripped-off" by industry parasite who have little or no ability to develop a product of their own so we can sympathize with their dilemma. All C.S.S. products have built-in safe guards which prohibit their use for flagrant pirating. The "IMPOSSIBLE"! is no exception! While The **"IMPOSSIBLE"!** backup the most heavily protected programs, it also checks to see that the 4K STATIC RAM pack is installed before allowing the backup copy to execute!

EXAMPLES: The **"IMPOSSIBLE"!** has been tested on 300 of the most popular and heavily protected programs we could find. With nearly 4000 programs for Atari, we DO NOT guarantee that it will backup all programs in the past-present-and future! We will supply updates at $6 each (non-profit!) if and when necessary. Programs we have successfully backed up include: Blue Max, Visi-cal, Archon, Mule, File Manager 800 +, Syn Calc, Syn File, One on One, 7 Cities of Gold, Super Bunny, Load Runner, Drol, and Gumball just to name a few!

Mastercard-Visa-Money Orders or Cashier Check. Phone: **(716) 467-9326** Please specify computer model number!

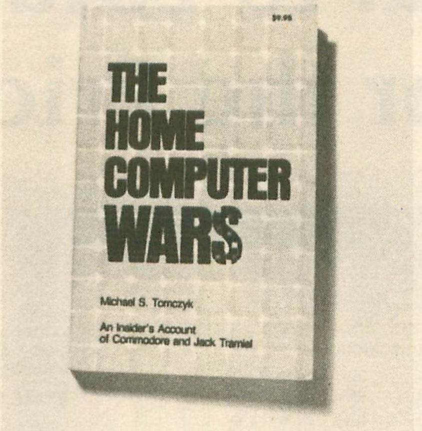
Send $149.95 plus $4 shipping and handling (N.Y.S. residents please add 7%)

COMPUTER SOFTWARE SERVICES
P.O. BOX 17660
ROCHESTER, N.Y. 14617

REVIEWS
continued from page 75

modules, and then program each module separately. This method is commonly used by advanced programmers and makes it possible to test and debug a program while you are writing it.

Appendices at the end of the book cover Error Messages, ATASCII codes, (numeric values for each character produced by the Atari), PEEK and POKE accessing of memory locations, and a display chart of the different graphics and text modes.



HOME COMPUTER WARS

by Michael S. Tomczyk
Compute! Publications, Inc.
P.O. Box 5406
Greensboro, NC 27403
(919) 275-9809
301 pages, paperbound, 1984
$9.95

Reviewed by Scott Lewis

The Home Computer Wars is a must read for any dedicated Atari fan. An insider's story of Commodore's successes in the home computer market, this book gives a detailed portrait of the current head of Atari, Jack Tramiel. It also provides a succinct analysis of the problems that haunted Atari during its transition from game machines to computers.

Much of the book is devoted to describing Jack Tramiel's business philosophy, referred to as "The Religion." The central tenet of this creed is, "Business is like war." Near the end of the book, the difference between the old and the new Atari is summed up by one insider: "Warner Communications likes to do things right. Jack likes to do things that work."

There are many more hints of what Tramiel might have in store for Atari hidden away in this book. To give one concrete example: During his first job interview with Tramiel, Tomczyk told him, "Your user manuals look like they're mimeographed. Apple has a two-color, spiral-bound booklet." At **Antic** just the other day we unpacked our first 130XE. The accompanying user booklet was printed in red and black, and, yes, it was spiral-bound.

Welcome to Home Computer War II.

COMPUTER TITLE BOUT

Avalon Hill
4517 Harford Road
Baltimore, MD 21214
(301) 254-9200
$30, 48K disk

Reviewed by Dr. John Stanoch

I was suffering from wargame "combat fatigue" when a friend told me about Avalon Hill's **Computer Title Bout**.

In this game of professional boxing, one or two players assume the roles of fight managers and decide what strategy their fighters should use in each round. As in real life boxing, victory is achieved by knocking out or outpointing the opponent in a 10, 12 or 15 round match.

One of the two disks contains the main game program and the other has data files for over 500 historical boxers, flyweights to heavyweights. You can also create your own fighter.

One night, I treated myself to a card that matched Sugar Ray Leonard against Roberto Duran, Chuck "Bleeder" Wepner vs. Jack Dempsey and Rocky Marciano vs. Muhammad Ali. . . all in their prime!

Each fighter is rated in some 20 categories such as aggressiveness, endurance, defense, hitting value, and effectiveness when facing a boxer or slugger. Fighters are controlled by issuing them a fighting "order" at the beginning of a round.

This order consists of a single number and letter. First is the strategy such as fighting inside, fighting outside, covering up and going for the knockout. Each strategy type can be used by a fighter only a limited number of times during the match. If the fighter is issued a strategy type more frequently than allowed, the computer will change it to NO STRATEGY.

The second part of the order is the fighter's attitude, ranging from an all-out attack to an all-out retreat.

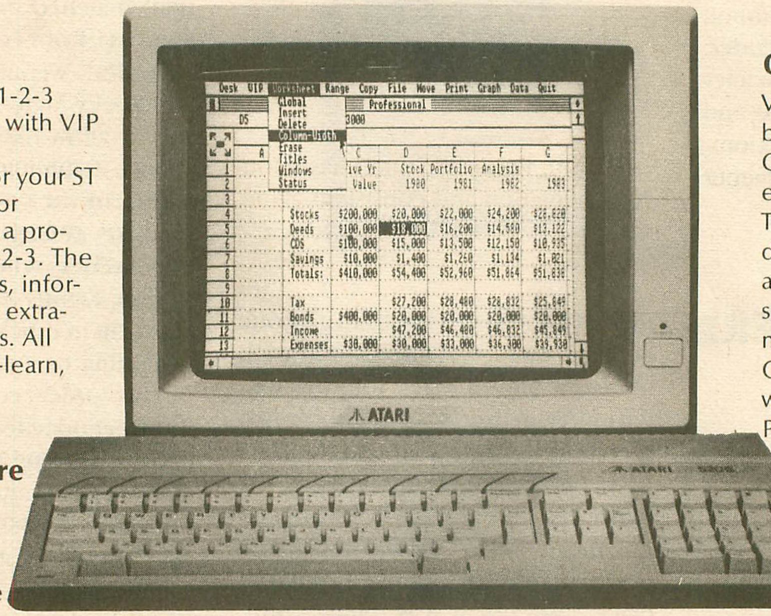
After the orders are given by both sides, the screen flashes to the execution phase in which the boxing ring is displayed. The fighters, ring and referee are portrayed in stick figures.

The time remaining in each three minute round and the round number are shown in the upper corners of the screen. At the bottom of the screen there is a two-line text window which lists what a particular fighter is saying and what type of punch he is throwing. A fighter may say, "You are my punchbag" or "Feel this pain" as he delivers a combination. The next moment, an opponent may retort with "Your mama" and a right uppercut.

continued on page 79

VIP Professional™

For Your Atari ST™ Identical to Lotus 1-2-3™ Except for the Price: $99.95



A True 1-2-3 Clone

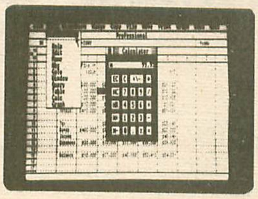
1-2-3 power without the 1-2-3 price. That's what you get with VIP Professional, the premier integrated spreadsheet for your ST from VIP Technologies. For $99.95, your ST can have a program identical to Lotus 1-2-3. The same spreadsheet analysis, information management and extraordinary business graphics. All combined in one easy-to-learn, easy-to-afford package.

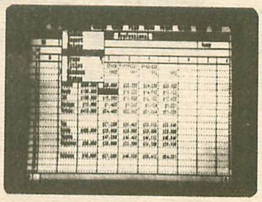
VIP Professional More Than Compares!

VIP Professional not only has all the features of 1-2-3, you can also type the same keys to do the same things. Or better yet, use a mouse to make it all so much simpler. And, of course, any off-the-shelf application made for 1-2-3 can be used with VIP Professional too!

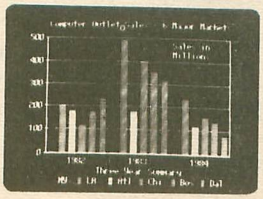
But VIP Professional goes on to teach 1-2-3 some new tricks. Just as the ST outstrips the power and performance of the business computers, VIP Professional meets and outfeatures Lotus. Compare for yourself:

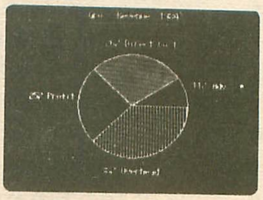
Feature	VIP	1-2-3
123 Commands	Yes	Yes
123 Graphics	Yes	Yes
123 Database	Yes	Yes
123 Macros	Yes	Yes
123 Files	Yes	Yes
16 colors	Yes	No
Uses mouse	Yes	No
Drop-Down Menus	Yes	No
Icons	Yes	No
GEM interface	Yes	No
Multiple windows	Yes	No
Available on ST	Yes	No
Easy to use	Yes	No
Affordable	Yes	No





VIP Professional will do your finances, keep your inventory, graph your sales, and a whole lot more.





One of ATARI's new GEMS

VIP Professional is easy to use because it takes advantage of the GEM™ environment and other elegant features built into the ST. The touch of your mouse, icons, and drop-down menus all make using it as easy as can be. Zip around the sheet, draw a graph, or issue a command with the click of your mouse. Or use the well-designed keyboard, whatever suits your fancy. And VIP Professional takes your figuring over the rainbow of colors that come with the ST, and uses the high-res screen for optimum graphics display.

Throw in lots of help, a thoughtful tutorial for the novice, and a thorough reference manual, and you have the makings of the perfect powerhouse product for the office, school and home. And at the super low price, VIP Professional is as easy to afford as it is to use.

Although VIP Professional will sell for $149.95 after November 1, as a special introductory offer, you can get this product for the low price of:

MasterCard **$99.95** VISA

To order direct call:

1-800-328-2737

In California call:

1-800-468-8737

Order status and software questions call 805-968-4364. Mail orders: For shipping and handling add $3.00 in the U.S., $6.00 Canada and $20.00 overseas. Personal Checks allow 3 additional weeks. Prices subject to change without notice.

VIP

Technologies

132 Aero Camino, Santa Barbara, CA 93117

REVIEWS
continued from page 77

In this phase, both players have no direct control over their fighters, and must, as any real life manager, watch from ringside and shout encouragements.

This game gives you the excitement and challenge of real professional boxing without the crowds, noise or blood. I immensely enjoyed the tactical options, easy rules, and huge array of fighters to chose from. Computer Title Bout is definitely a sleeper of the year.

BATTLE OF SHILOH

Strategic Simulations, Inc.
883 Stierlin Rd., Building A-200
Mountain View, CA 94043
(415) 964-1200
40K disk, 32K cassette
$39.95

Reviewed by Robert Fox

Battle of Shiloh is an interesting war simulation that plays well, but falls short of being memorable.

The game re-enacts the well-known Civil War battle, pitting the Union armies of Ulysses S. Grant against Confederate armies commanded by General Albert Johnston. Both sides use infantry and artillery units to gain control of Pittsburgh Landing—the key to winning the campaign.

The scenario is intriguing, and Shiloh possesses many attributes that further enhance its playability. Players can take command of either side and alter the relative strength of the armies from what they historically possessed. Terrain is important and troop movements require careful planning. SSI was wise in making Shiloh a one-or-two player game, since two players can initiate more spirited and unpredictable battles than those of the computer.

Shiloh does, however, have several weaknesses. The graphics are murky, making it difficult to distinguish between forest, field and hilly terrain. The computer calculates a simple random loss factor for each skirmish. This means that a surrounded and heavily outgunned army can still inflict unrealistically heavy losses on its adversaries! The main disapointment, though, is in the game's complete lack of sound, a feature that could have contributed greatly to the player's enjoyment.

Overall, Battle of Shiloh has good documentation and consistent game play, but lacks the extra creative effort that might have made it a classic.

BROADSIDES

Strategic Simulations, Inc.
883 Stierlin Rd, Bldg. A-200
Mountain View, CA, 94043
$39.95, 48K disk

Reviewed by Dr. John Stanoch

Think of fighting in the "Age of Sail" and you envision majestic vessels moving gracefully over a balmy sea beneath a canopy of flowing white sail. But these beautiful ships were really lethal war machines bristling with the most advanced weaponry of the day. With SSI's **Broadsides**, you can experience the tension and challenge of wind-powered warfare.

Broadsides places one or two players at the helm of "tall ships" around the latter part of the 18th century. Victory is achieved by sinking, capturing, or outscoring the opposing ship within 12 hours of game-time. (In real-world time, an average game lasts 30 minutes or less.)

There are three levels of play: the arcade version, the boarding screen only version, and the full tactical game that encompasses all the rules and options available.

Two of the screens are sailing modes, one at 2,400 yards across, the other at 600 yards across. They depict a topdown view of both ships with the wind direction and velocity, score and screen scale displayed at the bottom. The boarding screen is displayed when the two ships are grappling.

To the side of the playfield is a silhouette of each ship with its damage, maximum and current speed, number of remaining guns and sails. Ships are controlled via joystick or keyboard when on a sailing screen, and by only keyboard commands during grappling.

One of the outstanding characteristics of this game is the graphic details. For example, when a salvo misses the opposing ship a water spout is displayed. Crewmen fall when hit by sniper fire, during swordplay the clash of steel is heard in the background. The ship's sails become inreasingly riddled as they take damage.

The speed of play can be changed anytime while on the sailing screen. This allows a new player to slow the game down. Advanced players are given the option of designing their own ships from stem to stern. Eleven historic scenarios are provided with the game the player may change any of these parameters too.

One of the few faults I can find is that nowhere in the rules is the effect of wind direction and velocity on the play of the game explained. Any captain of that age had to thoroughly understand the proper use of the wind in order to defeat an enemy ship. While playing the game, I found that the speed of a ship drops to zero knots when it's turned into the wind.

There is only one ship per side, so this game does not simulate large fleet action. However, the superb graphics succeed overwhelmingly in providing a realistic and playable simulation of a duel between two wooden ships. I recommend Broadsides to *any* computer gamer.

continued on next page

ALLEY CAT

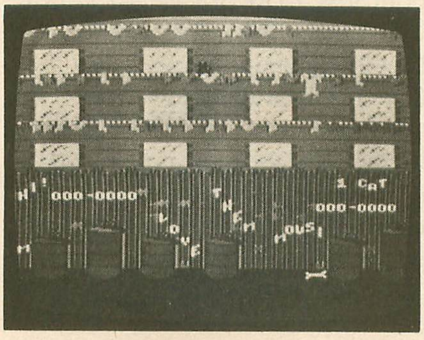
Synapse Software
17 Paul Drive
San Rafael, CA 94903
(415) 479-1170
$19.95, 48K disk

Reviewed by Scott Lewis

A 19th century wit once remarked that the entire tragedy of Romeo and Juliet was not equal to the story of a cat in heat. After playing **Alley Cat** you might very well end up agreeing with him.

As Freddy, an alley cat seeking the love, etc., of fair Felicia, you must run a gauntlet of obstacles to even come within kissing range.

First of all, you have to get into the Catalina Condominiums, Felicia's residence. Jumping through the windows is difficult enough, what with an angry dog and a stool-pigeon tramp of a cat lurking below. But once inside your ordeal has just begun.



Depending on which room you jump into (a completely random factor), you may be asked to catch four mice in a huge piece of cheese, drink the milk right out from under the noses of six sleeping dogs, or dive into a bowl of water and eat twelve goldfish without touching the bluely evanescent electric eels.

The graphics are really superb. Concentrating on the everyday (well, almost) instead of bizarre occurences in outer space, Bill Williams has created a challenging and interesting game with something of a storybook quality to it. Each new graphics scenario is a pleasure to see, and the animation of Freddy himself is completely first-rate. Freddy's main method of getting about is jumping, and owners of kittens will be pleasantly suprised by Williams' realistic duplication of "cat-in-flight."

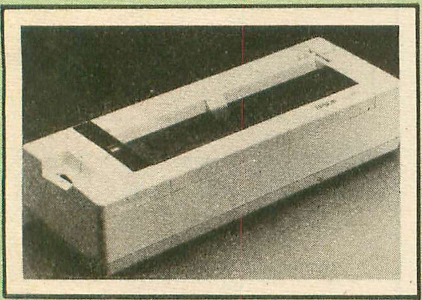
The game has several difficulty levels, and becomes automatically harder as you continue playing. Score is kept for you, but it doesn't really seem to matter. Alley Cat is simply a lot of fun to play!

ADVERTISERS

We encourage you to patronize our advertisers—all of whom support the Atari computers. We will appreciate your mentioning **Antic** when you contact these firms.

This is provided as a convenience and as a courtesy to advertisers. ANTIC does not guarantee accuracy or comprehensiveness.

new products



EPSON DX-10

(printer)
Epson America, Inc.
2780 Lomita Boulevard
Torrance, CA 90505
(213) 539-9140
$399

Epson debuted several new printers at the Comdex trade show. The **DX-10** is an 82-column daisywheel printer featuring bidirectional logic seeking printing. It uses a standard Diablo 96 character printwheel and Olivetti ET-121 ribbon.

The **HomeWriter 10** is an 80 column dot-matrix printer with a plug-in cartridge that makes it compatable with the Atari. It produces draft quality printing at 100 cps and near letter quality at 16 cps. Numerous type styles are available through Epson's SelecType feature. The HomeWriter is priced at $249, and the plug-in cartridge is $50.

The **HS-80 Letterjet** is a nine nozzle ink jet printer that weighs just an ounce over four pounds. The battery portable unit prints 80 columns at 160 cps draft and 32 cps near letter quality. Retail price is $449.

BASIC TUTORIAL FOR ATARI

(software)
DP Software
104 Barrymore Boulevard
Franklin Square, NY 11010
(516) 352-5605
$29, 32K, 2 disks

This is a comprehensive disk tutorial on Atari BASIC. It features a comprehensive syntax reference guide and over 300 different screens. A very user friendly tutorial, requires only four keys to operate the program.

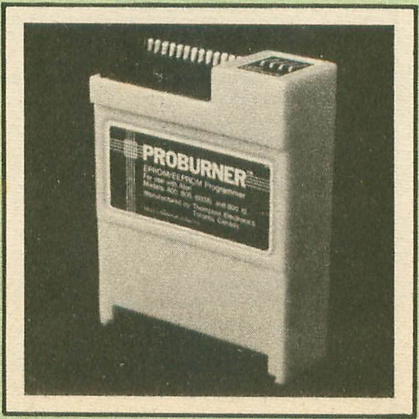
New Products notices are compiled by the **Antic** staff from information provided by the products' manufacturers. Antic welcomes such submissions, but assumes no responsibility for the accuracy of these notices or the performance of the products listed.



TEL-A-MODEM

(modem)
Code-A-Phone Corp.
16261 S.E. 130 Avenue
Clackamas, OR 97015
(503) 655-8940
$595

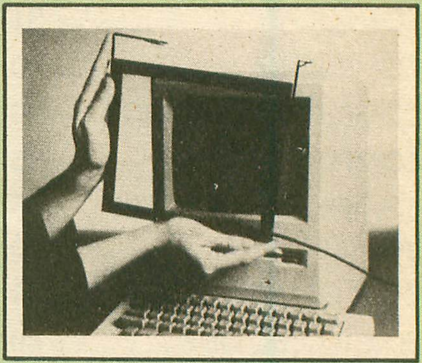
If you've found yourself running out of desk space lately, the **Tel-A-Modem** may well be for you. This versatile device combines a telephone, a modem and the power supply all into one neat package. Voice and data communications can be combined, as can tone and pulse dialing. 300 or 1200 baud transmission speeds can be selected.



PROBURNER

(EPROM programmer)
Thompson Electronics
1074 Kensington Avenue, Suite 188
Buffalo, NY 14215
(416) 960-1089
$149, 32K cartridge

This is a professional quality EPROM burner in a cartridge that plugs into your Atari. It features copy, erase check, verify and burn-in functions, as well as the capability to run chips from its socket. It can be used to program EPROM types 2716 to 27128, the 2532 and a variety of 2K through 8K EEPROMS.



VDT BODYGUARD

(glare protector)
Sentinel Bio-Tech Products
One Sentinel Plaza
Hyannis, MA 02601
(617) 775-5220
$164

In addition to being an anti-glare, antireflection and anti-static screen, the manufacturer claims that the **Bodyguard** also blocks 98% of emitted radiation. The very fine wire mesh screen is clipped onto the front of the terminal screen and held in place with an elastic strap.

Return the favor. When you call a manufacturer or supplier about a product you've seen advertised or otherwise mentioned in ANTIC, please tell them so. This will help us to continue to bring you the latest information about products that will make your Atari computer an even more valuable investment in the future. —ANTIC ED